



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV INTELIGENTNÍCH SYSTÉMŮ
FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF INTELLIGENT SYSTEMS

SLEDOVÁNÍ POHYBŮ OSOB V ZÁZNAMU ZE STACIONÁRNÍ KAMERY

PEDESTRIANS TRACKING IN A VIDEO RECORD FROM A STATIONARY CAMERA

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

MILAN TRNKAL

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. TOMÁŠ GOLDMANN

BRNO 2016

Zadání bakalářské práce

Řešitel: **Trnkal Milan**

Obor: Informační technologie

Téma: **Sledování pohybů osob v záznamu ze stacionární kamery**

Pedestrians Tracking in a Video Record from a Stationary Camera

Kategorie: Zpracování obrazu

Pokyny:

1. Seznamte se s problematikou detekce osob ve videu ze stacionární kamery. Dále se seznamte s možnostmi sledování pohybu detekovaných osob.
2. Analyzujte a sumarizujte dostupné metody pro detekci a sledování pohybujících se osob ve videu.
3. Navrhněte algoritmus pro detekci a sledování pohybu osob se záznamem trajektorií pohybu.
4. Algoritmus implementujte v libovolném programovacím jazyce s důrazem na výpočetní rychlost řešení.
5. Algoritmus otestujte na množině videí z městského prostředí. Zhodnoťte spolehlivost navrženého řešení, zhodnoťte výsledky a navrhněte možnosti pokračování projektu.

Literatura:

- Sonka, M., Hlavac, V., Boyle, R.: *Image Processing, Analysis, and Machine Vision*. Cengage-Engineering, 2007.
- Davies E. R.: *Computer and Machine Vision, Fourth Edition: Theory, Algorithms, Practicalities*. Academic Press, 2012.

Pro udělení zápočtu za první semestr je požadováno:

- Bez požadavků.

Podrobné závazné pokyny pro vypracování bakalářské práce naleznete na adrese

<http://www.fit.vutbr.cz/info/szz/>

Technická zpráva bakalářské práce musí obsahovat formulaci cíle, charakteristiku současného stavu, teoretická a odborná východiska řešených problémů a specifikaci etap (20 až 30% celkového rozsahu technické zprávy).

Student odevzdá v jednom výtisku technickou zprávu a v elektronické podobě zdrojový text technické zprávy, úplnou programovou dokumentaci a zdrojové texty programů. Informace v elektronické podobě budou uloženy na standardním nepřepisovatelném paměťovém médiu (CD-R, DVD-R, apod.), které bude vloženo do písemné zprávy tak, aby nemohlo dojít k jeho ztrátě při běžné manipulaci.

Vedoucí: **Goldmann Tomáš, Ing.**, UITS FIT VUT

Datum zadání: 1. listopadu 2015

Datum odevzdání: 18. května 2016

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
Fakulta informačních technologií
Ústav inteligentních systémů
602 00 Brno, Božetěchova 2

doc. Dr. Ing. Petr Hanáček
vedoucí ústavu

Abstrakt

Tato práce se zaměřuje na problematiku sledování chodců kamerou. Představil jsem používané metody počítačového vidění, které jsou vhodné pro detekci a klasifikaci osob. Na základě detekce pohybu osob ve videozáznamu jsem navrhl algoritmus pro detekci a sledování chodců. Výsledná aplikace používá histogram orientovaných gradientů a SVM klasifikátor, pro identifikaci chodců jsou využity barevné histogramy. Zjištěné trajektorie jsou následně vykreslovány na výstup. V poslední části práce se zabývám testováním a vyhodnocením výsledků navrženého algoritmu.

Abstract

This bachelor thesis focuses on pedestrians tracking from camera. In this work, I have introduced several methods of computer vision suitable for detection and classification of people. I proposed an algorithm for detecting and tracking pedestrians based on detection of movement. The application uses a Histogram of Oriented Gradients and SVM classifier together with color histograms for identification of pedestrians. Pedestrian's trajectories are then rendered to the output. Last part of the thesis deals with testing and evaluation of the results of the algorithm.

Klíčová slova

Histogram gradientů, SVM, Extrakce příznaků, Adaptivní směsice Gaussových křivek, Detekce lidí, Sledování chodců, OpenCV, Detekce pohybu ve videu, Záznam pohybu

Keywords

Histogram of Oriented Gradients, SVM, Feature extraction, Gaussian mixture model, People detection, Pedestrian tracking, OpenCV, Detection of movement in video, Trajectory capture

Citace

TRNKAL, Milan. *Sledování pohybů osob v záznamu ze stacionární kamery*. Brno, 2016. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Goldmann Tomáš.

Sledování pohybů osob v záznamu ze stacionární kamery

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Tomáše Goldmanna. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Milan Trnkal
17. května 2016

Poděkování

Chtěl bych poděkovat panu Ing. Tomáši Goldmannovi za kvalitní odborné konzultace a rady, které napomohly k úspěšnému dokončení této práce. Zároveň bych chtěl poděkovat všem, kteří mě během práce jakkoliv podpořili.

© Milan Trnkal, 2016.

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

1	Úvod	3
2	Sledování chodců	4
2.1	Satelitní technologie	4
2.1.1	GPS	4
2.1.2	GLONASS	5
2.1.3	Využitelnost pro sledování chodců	5
2.2	Video kamery	5
2.2.1	CCD kamerové čipy	6
2.2.2	CMOS kamerové čipy	7
2.2.3	CCD, CMOS a využitelnost pro sledování chodců	7
2.2.4	Termovize	8
2.3	Metody pro sledování chodců kamerou	9
2.3.1	Sledování pomocí tepelných stop	9
2.3.2	Sledování pomocí siluet	9
2.3.3	Sledování na základě detekce pohybu	10
3	Zpracování obrazu	11
3.1	Základní metody pro zpracování snímků	11
3.1.1	Metoda rozdílu snímků	11
3.1.2	Metoda využívající mediánový filtr	12
3.1.3	Metoda směsice Gaussových křivek	13
3.1.4	Dilatace	15
3.1.5	Eroze	15
3.2	Segmentace obrazu	16
3.2.1	Odčítání pozadí	16
3.3	Detekce a popis oblastí zájmu	18
3.3.1	Detekce binárního objektu	19
3.3.2	Scale-Invariant Feature Transform	20
3.3.3	Speede-Up Robust Features	22
3.3.4	Histogram of Oriented Gradients	23
3.4	Klasifikace objektů	25
3.4.1	Adaptive Boosting	25
3.4.2	Support Vector Machine	26

4	Návrh a implementace aplikace	28
4.1	Použité nástroje	29
4.2	Vstupní data	29
4.3	Zpracování snímku	29
4.3.1	Úprava snímku	29
4.3.2	Detekce objektů v popředí	30
4.3.3	Detekce blobů	30
4.3.4	Detekce osob	31
4.3.5	Sledování osob	31
4.3.6	Záznam trajektorie sledované osoby	35
4.4	Výstup	35
5	Testování	36
5.1	Sledování chodců v reálném prostředí	36
5.1.1	Detekce a klasifikace	36
5.1.2	Sledování chodců	37
5.1.3	Zhodnocení	37
5.2	Sledování chodců v kontrolovaném prostředí 1	38
5.2.1	Detekce a klasifikace	38
5.2.2	Sledování chodců	38
5.2.3	Zhodnocení	39
5.3	Klasifikace a rychlost sledování	39
5.4	Testovací sestava	40
6	Závěr	41
	Literatura	43
A	Obsah CD	47

Kapitola 1

Úvod

Počítač, který si člověk zkonstruoval, se za posledních několik desítek let stal neodlučnou součástí našeho každodenního života. V současné době počítače dovedou v mnoha aspektech člověka i zastat, ale mají zásadní nedostatek. Nedokáží vnímat svět tak jako lidé. Zrak je jeden ze základních smyslů člověka, který nám umožňuje přijímat podněty ze světa kolem nás a na jejich základě rozhodovat. Stroje naopak tento smysl až do nedávné historie postrádaly.

Jeden z hlavních důvodů, proč zpracování obrazu stálo mimo široký zájem společnosti, byl nedostatečný výpočetní výkon. Stroje neměly dostatek výkonu pro velmi složité a náročné operace, které jsou používány při zpracování obrazu. Další překážkou byly zdroje obrazových dat. Kamery byly do přelomu druhého tisíciletí převážně analogové a kvalita výstupu běžných kamer nebyla velká. Cena těchto záznamových zařízení byla také vysoká, což bránilo většímu rozšíření.

Tento trend se zlomil právě kolem roku 2000. Začaly se objevovat počítače s dostatečným výkonem a záznamová zařízení, které byly schopny dodávat kvalitní záznamy. Najednou bylo možné použít stroje i na úkoly, které do té doby byly mimo jejich možnosti. Jednou z možností použití jsou kamery na výrobní lince, které jsou schopné pomocí počítačového vidění rozpoznat defektní výrobky a to i ve velmi vysoké rychlosti. Po množství velmi nešťastných a násilných událostech počátkem tisíciletí, se začalo počítačové vidění používat pro rozpoznávání osob a jejich monitorování. Začaly se vyvíjet velmi sofistikované systémy pro sledování osob a analýzu jejich chování v rámci zajištění bezpečnosti.

Zde popsaná práce se zabývá detekcí chodců a jejich sledováním kamerou. Kamerou je v této práci chápáno záznamové zařízení umožňující pořizování kontinuálního video záznamu, které je umístěno ve výšce nad sledovanou oblastí. Může se jednat o bezpečnostní kameru nebo kameru umístěnou do dané oblasti za účelem sběru informací o pohybu osob. Cílem práce není identifikace jednotlivých osob, ale zaznamenání tras, kde se chodci v dané oblasti nejčastěji pohybují. Výstup této aplikace je pak možné použít pro případné fyzické úpravy sledované oblasti. Tyto úpravy mohou být například přesun objektů (lavička, odpadkové koše, stánky), které narušují plynulý provoz pro chodce nebo úplná přestavba celého území. Lze takto sledovat i zajímavé jevy, jako je chování lidí na různé situace a tyto informace poté použít při budoucím plánování nových městských částí.

Práce se tedy zabývá analýzou současných řešení pro sledování chodců 2 a rozebírá dnešní možnosti počítačového vidění. V části, kde je rozebíráno zpracování obrazu 3 jsou popsány jednotlivé kroky a metody jak je vyřešit. Vybrané algoritmy jsou pak použity pro řešení aplikace pro sledování chodců kamerou. Nakonec dochází k uchování a vykreslení trajektorií jednotlivých chodců.

Kapitola 2

Sledování chodců

Tato kapitola je zaměřena na již existující řešení pro sledování pohybu osob v prostoru. Pokusím se popsat používané metody a možnosti jejich praktického využití. V dnešní době jsou řešení problému sledování osob velmi rozmanitá od bez kamerových systémů založených na čidlech, přes satelitní čipy v oblečení nebo mobilních telefonech až po kamerové systémy. U metod využívající čidla se jedná zejména o optické závory a PIR¹ snímače, které slouží pro detekci narušení sledovaného prostoru. Optické závory pracují na principu detekce přerušení světelného paprsku, který vyzařuje emitor. PIR čidla naopak detekují záření které se vyskytuje v jejich dosahu a pomocí filtrů se snaží detekovat záření o vlnových délkách 8 μm až 14 μm , které odpovídají spektru vyzařovaným lidským tělem. Může docházet ke kombinaci obou typů čidel za účelem zlepšení detekce pohybu, zvláště je vhodné tyto typy čidel kombinovat při snaze o pokrytí velkých ploch nepravidelných tvarů. Jejich použití cílí převážně na počítání chodců a detekci narušitelů, pro sledování chodců jsou tedy nevhodné.

Největší technologický posun se ovšem týká kamerových systémů, které se za posledních 10 let rozvinuly a rozšířily v oblasti sledování. Za tento rozvoj lze především vděčit rapidnímu nárůstu výpočetnímu výkonu a jeho nízké ceně.

2.1 Satelitní technologie

Jedná se o navigační systémy založené na satelitech na oběžné dráze Země. V dnešní době jsou nejdominantnější dva systémy, každý spoléhající na rozdílné způsoby přenosu signálu. Tyto systémy jsou GPS² a GLONASS³. Další je například nedokončený Evropský civilní navigační systém Galileo[16] nebo čínský navigační systém BeiDou[4].

2.1.1 GPS

Jedna z nejrozšířenějších technologií pro sledování osob a majetku je založena na satelitní technologii vyvinutou pro americkou armádu během 20. století[26]. V současné době, je na oběžné dráze Země ve výšce okolo 20 200 km [30] v provozu 31 satelitů[18] umožňující navigaci. Každý satelit nese atomové hodiny, které jsou synchronizovány mezi jednotlivými satelity na oběžné dráze a pozemním střediskem. Základní předpoklad pro správnou funkci GPS je předem známý přesný čas a pozici jednotlivých satelitů.

¹Passive Infrared Detector

²Global Positioning System

³GLOBAL NAVIGATION SATELLITE SYSTEM

Každý satelit nestále vysílá modulovanou vlnu, signál, který obsahuje čas vyslání a pseudo-náhodný kód, který je známý všem přijímačům. Pro samotné zjištění polohy je potřeba detekce signálů ze čtyř různých satelitů systému GPS. Tři satelity pro určení polohy a jeden pro synchronizaci času. Ve speciálních případech je možné použít signálů z méně satelitů, pokud některé z hodnot jsou předem známé, například letová výška letadla nebo nadmořská výška plavidla. Zařízení určená pro civilní sektor umožňují pouze pasivní příjem tohoto signálu, neexistuje tedy žádná možnost jak z těchto satelitů monitorovat jednotlivé přijímače.

2.1.2 GLONASS

Satelitní systém GLONASS byl stejně jako GPS 2.1.1 vyvinut pro vojenské účely v 70. letech 20. století. Oficiálně, plného globálního pokrytí pomocí 24 operačních satelitů dosáhl v roce 2011[17]. Satelity jsou umístěné o něco níže, než konkurenční GPS, a to ve výšce 19 140 km[27]. Hlavní rozdíl oproti GPS je v použitém signálu a jeho struktuře. GLONASS používá signál s frekvenčním dělením kanálu. Satelity tohoto systému, stejně jako GPS, vysílají dva typy signálů, civilní a vysoce přesný vojenský. Vysílání probíhá na 15 frekvenčně dělených kanálech pro každé ze dvou frekvenčních spekter. Civilní signál vysílaný těmito satelity se skládá z pseudo-náhodného kódu, navigační zprávy a záložní sekvence. Stejně jako GPS, tento systém neumožňuje přímé monitorování polohy přijímačů.

2.1.3 Využitelnost pro sledování chodců

Jak bylo zmíněno, zařízení postavené na příjmu signálu z GPS nebo GLONASS samy o sobě neumožňují aktivní monitorování osob. Z tohoto důvodu jsou tato zařízení často vybavena dalšími jednotkami, které umožňují vysílání zjištěných údajů bez závislosti na satelitech, tyto jednotky jsou obvykle GSM⁴ moduly.

Ve většině případů se jedná o menší zařízení vybavené malou anténou, příjem v místech s omezeným výhledem na oblohu je tedy velmi komplikovaný a často i nemožný. Tento palčivý nedostatek vyřešili některé firmy spojením satelitního přijímače s velmi citlivými snímači jako jsou akcelerometry, gyroskopy nebo barometry. Tyto senzory pak umožňují sbírat přesná data o poloze osoby i v případě výpadku satelitního signálu. Jedno z řešení nabízí firma PNI a jejich SENtrace čip[31], který lze vsunout do obuvi a tím zajistit neustálé monitorování pohybu.

Systémy založené na GPS nebo GLONASS jsou velmi přesné a spolu s dalšími nástroji dokáží velmi přesně monitorovat pohyb osob kdekoli na Zemi. Pro účely sledování neznámých osob jsou ovšem špatně použitelné. Každá sledovaná osoba by musela být vybavena velmi drahým kompatibilním zařízením, který by umožňoval sledování jejich pohybu. Tento systém je tedy vhodný pouze pro sledování známých osob např. řidičů, skladníků apod.

2.2 Video kamery

Základní struktura všech kamerových systému je velmi podobná. Pro získání snímku kamerou musí paprsky světla projít systémem čoček a dopadnout na světlocitlivý člen kde dochází k vyhodnocení informací a vytvoření výstupního snímku. Čipy kamerových systémů se obvykle skládají z filtru, samotného snímače a případně další řídicích obvodů potřebných pro správnou funkci. Snímač, podle použitého typu, detekuje příchozí záření a po zpracování

⁴GSM - Groupe Spécial Mobile - Globální Systém pro Mobilní komunikaci

specializovanými obvody je hotový snímek poslán na požadovaný výstup mimo kamerový modul (např. uložení na paměťový modul).

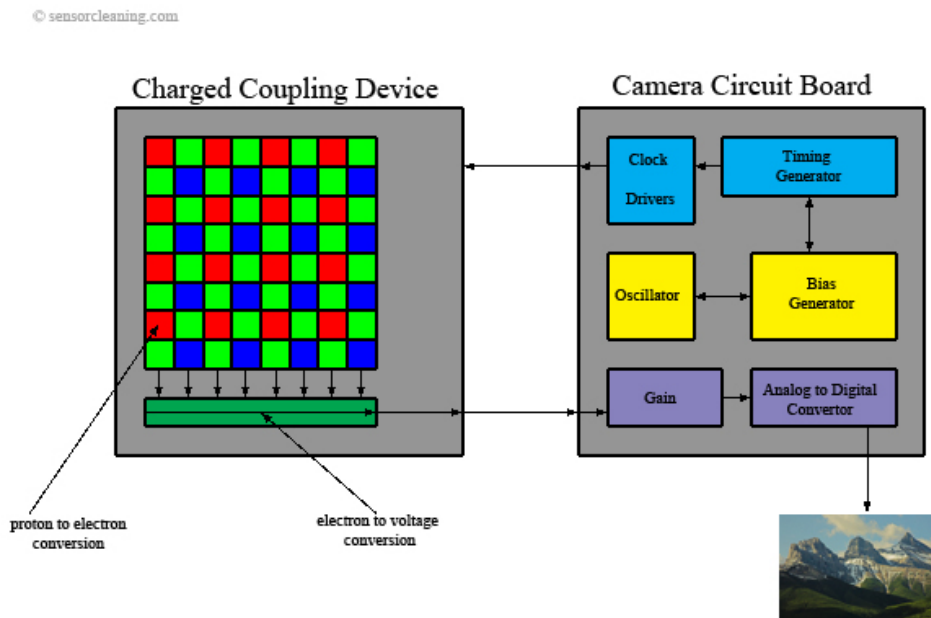
Před každým senzorem je potřeba barevného filtru, který umožňuje rozpoznat barvy. Bez barevného filtru jsou tedy výstupní snímky černobílé, jelikož je známa jen informace o přítomnosti světla, ne o jeho vlastnostech. Samotný čip se skládá ze světlo-citlivých buněk, pixelů, které detekují množství světla, které na ně dopadá. V závislosti na množství světla dopadající na daný pixel, se generuje náboj, který je převeden na napětí. Získané napětí se poté přes A/D převodníky mění na digitální signál.

V současné době jsou nejrozšířenější kamerové čipy postavené na dvou technologiích, CCD (Charge-Coupled Device) a CMOS (Complementary Metal-Oxide-Semiconductor). K rozšířeným kamerovým technologiím patří i termokamery.

2.2.1 CCD kamerové čipy

CCD snímač generuje z každého použitého pixelu náboj v závislosti na intenzitě dopadajícího světla. Náboje z jednotlivých pixelů jsou čteny řádek po řádku pomocí vertikálních a horizontálních registrů. Elektrický signál je poté zesílen na výstupním zesilovači a převeden do digitální reprezentace [1].

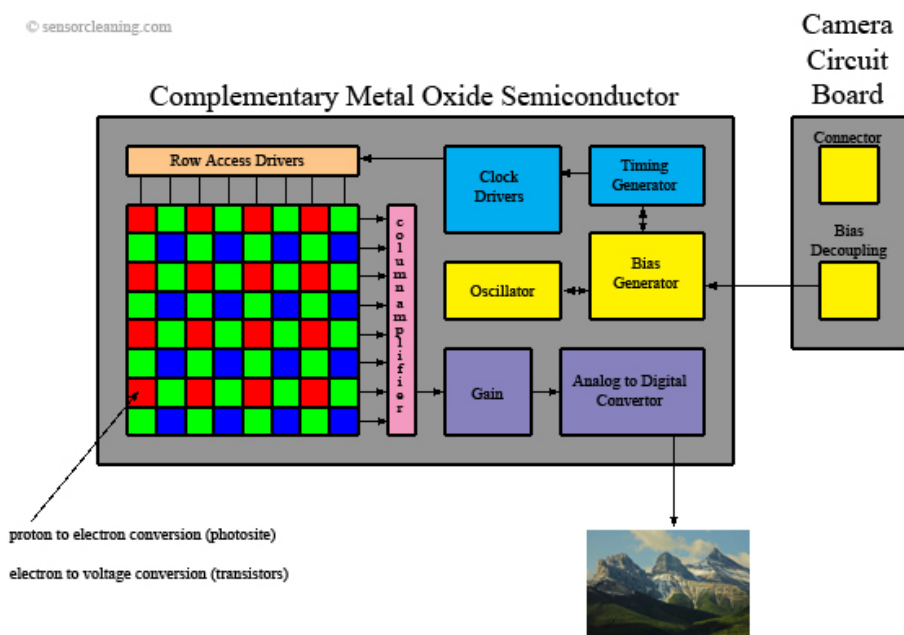
Podle způsobu čtení informací z jednotlivých buněk se čipy dělí na prokládané, progresivní a plošné. Prokládané čtení je známé především z oblastí televize a videa. Čtení probíhá nejprve v lichých řádcích, poté v sudých a až následně je skládán výsledný snímek. Celý proces získávání snímku je pomalý díky množství operací a způsobu čtení informací z jednotlivých buněk. Progresivní čtení spočívá v přečtení vždy celého řádku najednou, v pořadí ve kterém jsou uspořádány. Čtení je tak rychlejší a obraz ostřejší, je možné použít i elektronickou závěrku pro velmi krátké expoziční časy. Poslední způsob je plošné čtení. Při jeho použití je celý čip přečten najednou, je tak možné použít čip na snímání mnohem dříve než u předchozích variant.



Obrázek 2.1: Zjednodušené schéma CCD čipu. [6]

2.2.2 CMOS kamerové čipy

CMOS čipy, stejně jako CCD čipy, obsahují CCD elektrody a MOSFET tranzistory. V principu fungování se příliš neliší, největší rozdíly lze nalézt v architektuře čipu. Čipy postavené na CMOS technologii mají své zesilovače a A/D převodníky přímo integrované v čipu, není tedy potřeba dalších komponent mimo tento čip [1]. Je tím snížena cena, energetická náročnost a tím i teplota čipu. Každý pixel v čipu je postaven s obvody, které přímo převádí náboj na elektrické napětí. Díky vzájemné nezávislosti jednotlivých pixelů je možné číst i jen vybrané části čipu (výřez ze snímku) místo celé plochy. Tím je možné zvýšit počet snímků za vteřinu, které je čip schopný dodat. Lze také využít funkcí naklonění, natočení a přiblížení.

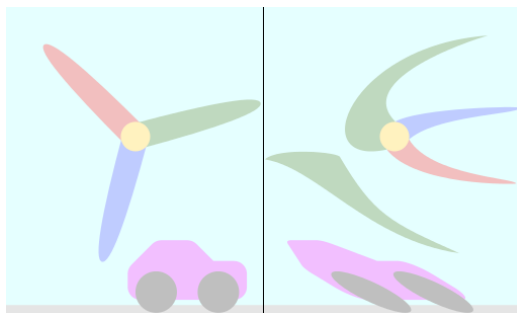


Obrázek 2.2: Zjednodušené schéma CMOS čipu. [6]

2.2.3 CCD, CMOS a využitelnost pro sledování chodců

CCD senzory jsou postaveny na mnoha analogových komponentách a pro některé tyto komponenty je potřeba využít obvody mimo čip. Tím se zásadně zvyšují energetické požadavky i výrobní náklady. Spolu s energií se zvyšuje i teplota celého zařízení. V důsledku se může zvýšit šance na selhání komponent, případně celého kamerového systému, ovlivněna může být ale i kvalita výstupního snímku. Mimo jiné se tím zvyšuje i cena provozu a údržby. Vliv těchto defektů je ale v rámci spotřebitelských výrobků zanedbatelný. Problémy pak nastávají při využití CCD čipu v zařízeních jako jsou velké teleskopy. Velká výhoda CCD čipu je možnost plošného čtení z čipu také označované jako total shutter. Oproti progresivnímu čtení, rolling shutter, je celý čip přečten najednou a nedochází tak k deformaci rychle se pohybujících objektů viz. obrázek 2.3.

Oproti CMOS čipům mají CCD senzory lepší výstupní snímky při zhoršených světelných podmínkách, tím se CCD technologie stává lepší volbou pro využití v bezpečnostních kamerách a tedy pro sledování chodců.



Obrázek 2.3: Rozdíl mezi plošným čtením (vlevo) a progresivním (vpravo) pro pohybující se objekty. [7]

2.2.4 Termovize

Termovize, stejně jako konvenční kamery probrané v 2.2.1 a 2.2.2, snímají elektromagnetické záření. Na rozdíl od konvenčních kamer se pracuje s podstatně delšími vlnovými délkami, až $14\,000\ \mu\text{m}$. Díky zvolené infračervené části spektra, není nutné ani žádné osvětlení z okolí. Každý objekt, který má teplotu vyšší než $0\ \text{K}$, vyzařuje v infračerveném pásmu. S teplotou objektu se také mění množství záření, které vydává. Hodnota kolik energie je každý objekt schopen vyzařit, se udává emisním koeficientem ε v rozmezí 0 až 1, kde 1 představuje objekt nevyzařující v infračerveném spektru. Tato zákonitost je využita pro termovize, kdy pro co nejjednoznačnější snímek je potřeba co největších teplotních rozdílů mezi jednotlivými objekty/části snímku.

Největší výhoda termovize je malá závislost na denní době nebo počasí. Protože tyto snímače pracují v infračerveném spektru, jsou výsledné snímky označovány jako termogramy v odstínech šedé barvy. Pro zlepšení čitelnosti a interpretace výstupních termogramů, je pořízeno několik stejných snímků skrz barevné filtry. Jejich kombinací lze získat barevně odlišené oblasti s různou teplotou.



Obrázek 2.4: Porovnání snímků z klasické barevné kamery (vlevo) a termovize (vpravo). [12]

Využitelnost termokamer pro sledování chodců

Díky způsobu, jakým termovize snímají danou oblast, se stávají vhodným kandidátem pro sledování osob. Kamerám nevadí den ani naprostá tma a narozdíl od konvenčních kamer se dokáží obstojně vypořádat i s nepříznivými přírodními podmínkami jako je mlha nebo déšť. Zásadní negativum této technologie je cena a dostupnost.

Termovize jsou velmi drahé v porovnání s tradičními kamerami a z důvodu možného vojenského použití se na ně vztahují přísná pravidla. Bez získání úřední registrace, lze zakoupit jen kamery s nízkým obrazovým rozlišením a snímkovou frekvencí. Taková kamera je tedy použitelná pro detekci pohybu, ale možnosti klasifikace objektů mohou být sníženy viz. 2.3.1. Pro nejlepší výsledky je vhodná kombinace termovize spolu s konvenční kamerou v jednom systému, kde si mohou navzájem asistovat. I přes značné výhody použití termovize, jsou pro svou cenu a dostupnost pro tuto práci nevhodné.

2.3 Metody pro sledování chodců kamerou

V této části práce se zaměříme na několik vybraných způsobů sledování objektů (osob), ve videozáznamech pořízených pomocí technologií popsanych v kapitole 2.2.

2.3.1 Sledování pomocí tepelných stop

Jak je uvedeno v sekci 2.2.4, termovize jsou schopné ve snímku detekovat objekty, které mají teplotu vyšší než je absolutní nula. Jako první krok je tedy potřeba provést detekci popředí. Jelikož v infračerveném spektru se nenachází stíny, je detekce popředí snadnější. V případě vhodných podmínek (např. chladná noc) jsou osoby oproti pozadí velmi výrazné. Za předpokladu, že lidské tělo má za normálních okolností teplotu mezi 36.5°C a 37°C , lze s určitou tolerancí ze snímku odfiltrovat oblasti nespadaající do daného rozmezí. Na získaných oblastech lze již provést klasifikaci na lidské postavy.

Bohužel, právě u klasifikace jsou tyto kamery ve velké nevýhodě kvůli nízkému rozlišení. Klasifikace chodců je tedy značně ztížena, zvláště pokud je založena na tvaru člověka, protože celé objekty/osoby mohou být reprezentovány jen několika málo pixely. Nejedná se o neřešitelný problém, ale je potřeba, aby objekty splňovaly tzv. Johnsonovy podmínky[20]. Tyto podmínky určují minimální velikosti daného objektu pro úspěšnou detekci, rozpoznání a identifikaci. Po úspěšném rozpoznání a případné identifikaci je poslední krok samotné sledování pohybu. V tomto kroku lze využít například některý z algoritmů pro predikci pohybu.

2.3.2 Sledování pomocí siluet

Sledování na základě siluet se snaží pomocí přesně popsanych objektů tyto objekty sledovat. Objekt v tomto případě může být část nebo i celé lidské tělo. Jedná se o složitý geometrický tvar, který nelze snadno popsat jednoduchými tvary. Jednotlivé části (např. končetiny) se hledají pomocí modelu, který je tvořen z předchozích snímků. Model, generovaný pro sledování siluet, může mít několik podob a to jako mapa hran, barevný histogram nebo tvar obrysu. K samotnému sledování je možné přistoupit z více stran.

Jednou z možností je sledování na základě obrysu objektu, který se upřesňuje v každém snímku za pomoci stavových prostorových modelů. Druhý přístup je sledování na základě tvaru objektu, kdy dochází k porovnání siluety z aktuálního snímku s referenčním modelem.

2.3.3 Sledování na základě detekce pohybu

Pro použití této metody se předpokládá, že sledovaná scéna obsahuje pohyb, v našem případě pohyb osob. Jedná se o metodu, která sama o sobě není schopná rozeznat, jaký objekt právě sleduje na rozdíl od předchozích metod.

Místo výpočetně náročné analýzy celého snímku na výskyt postav, se vytipují oblasti, kde se budou dané postavy vyhledávat. Za podmínky, že kamera je statická a hledáme postavy v pohybu, tak je možné vypočtením rozdílu mezi dvěma snímky zjistit, kde se vykonal nějaký pohyb. Takto lze získat binární masku, která odděluje statické pozadí od popředí. Aplikací masky na původní snímek jsou získány zájmové oblasti. Tyto oblasti jsou zpravidla mnohem menší, než celý snímek, čímž může být následná detekce urychlena. Nad zájmovými oblastmi se provede detekce a klasifikace na lidské postavy, které pak lze sledovat v rámci celého videozáznamu z kamery.

V případě sledování pomocí termovizí, jsme schopni nalézt osoby podle teploty, a při použití siluet můžeme mít předem připravený model chodce. Tyto možnosti metoda sledování pomocí pohybu neposkytuje, ale má několik výhod oproti ostatním přístupům. Je výpočetně méně náročná než sledování pomocí siluet, jelikož se provádí pouhý odečet hodnot jednotlivých pixelů ve snímku a není nutné porovnávat části obrazu s modelem pro nalezení podobných tvarů. Oproti termovizím má tato metoda podstatnou výhodu, že není závislá na použití drahých a provozně náročných kamer. Je možné použít libovolnou videokameru s dostačujícími parametry pro dané použití. Díky těmto výhodám, jsem se rozhodl, v této práci, pro použití metody sledování na základě detekce pohybu.

Kapitola 3

Zpracování obrazu

V této kapitole jsou popsány vybrané metody používané pro řešení detekce a sledování osob. Metody jsou rozřazeny do tří základních celků, a to zpracování obrazu, detekce objektů zájmu a sledování osob. Jde o základní části, které jsou s obměnami stejné pro většinu aplikací zabývajících se touto problematikou.

V první části rozebereme, jakým způsobem zpracovat obraz tak, aby bylo možné zjistit, zda je ve snímku nějaký pohyb. Druhá část je naopak zaměřena na lidské osoby. Pro každý zjištěný objekt je nutné rozhodnout, s jakou pravděpodobností se jedná o osobu a tedy, zda tento objekt bude dále zpracován. V poslední části je probrána problematika sledování detekovaných objektů.

3.1 Základní metody pro zpracování snímků

3.1.1 Metoda rozdílu snímků

Nejjednodušší metoda pro získání pozadí respektive popředí ve scéně. Pro výpočet masky M_t je použit snímek na vstupu v čase I_t a jako pozadí je určen snímek v čase I_{t-1} .

$$M_t = |I_{t-1} - I_t| \quad (3.1)$$

Jedná se o velmi primitivní metodu a výsledek nemusí dosahovat přijatelných výsledků, právě kvůli své jednoduchosti přístupu k detekci. Algoritmus pracuje s myšlenkou, že všechny pixely pozadí jsou statické, a naopak celé popředí se pohybuje. Pro vylepšení algoritmu lze použít prahovou hodnotu.

$$Threshold < |I_{t-1} - I_t| \quad (3.2)$$

Vzorec pro získání pozadí odečtem dvou snímků. [34]

Při odečtu jednotlivých pixelů mezi oběma snímky je použita prahová hodnota, jejíž pomocí se provede rozhodnutí, jestli v masce budou dané pixely použity. Práh je nutné zvolit tak, aby byly vyrušeny pouze nežádoucí obrazové změny.



Obrázek 3.1: Metoda rozdílu snímků a) původní snímek, b) maska získaná odečtení dvou po sobě jdoucích snímků.

3.1.2 Metoda využívající mediánový filtr

Jde o pokročilejší a náročnější metodu, než která je popsána v sekci 3.1.1. Algoritmus pracuje s modelem pozadí, který je počítán ze sledu několika po sobě jdoucích snímků, které se průměrují. Průměr je vypočten z hodnot všech pixelů na dané pozici s využitím median filtru [34].

$$B(x, y, t) = \text{median}\{I(x, y, t) - i\} \quad (3.3)$$

$$|I(x, y, t) - B(x, y, t)| > \text{Threshold} \quad \text{pro } i < 0, n - 1 > \quad (3.4)$$

Rovnice pro výpočet rozdílů mezi snímky pomocí mediánového filtru.

Model pozadí je postupně přepočítáván s každým krokem algoritmu, tím je zaručena možnost dobrého přizpůsobení změnám v obraze. Negativní dopad na výslednou masku může mít použití jednotného, a v čase neměnného, prahu.



(a)

Obrázek 3.2: Ukázka snímku po aplikaci mediánového filtru.

3.1.3 Metoda směsice Gaussových křivek

Metoda směsice Gaussových křivek, MoG¹[33], se zaměřuje na detekci změn ve snímku pomocí směsí Gaussových křivek. Metoda navrhuje adaptivní model Gaussových křivek pro každý pixel ve snímku zvlášť. Za předpokladu, že každý pixel je charakterizován určitou intenzitou na barevné škále RGB. Pak je pravděpodobnost výskytu právě vybraného pixelu dána rovnicí 3.5.

$$P(X_t) = \sum_{i=1}^K \omega_{i,t} \cdot \eta(X_t, \mu_{i,t}, \sum_{i,t}) \quad (3.5)$$

Rovnice pro výpočet pravděpodobnosti výskytu pixelu. [33]

Kde:

- K - počet distribucí,
- $\omega_{i,t}$ - váha i -tého Gaussiánu v čase t ,
- μ_i - střed Gaussiánu,
- \sum_i - kovariantní matice,
- η - Gaussova hustota pravděpodobnosti funkce 3.7.

Stauffer a Grimson předpokládali, že komponenty RGB jsou na sobě nezávislé a mají stejné variace [33].

$$\sum_{i,t} = \sigma_{i,t}^2 \cdot I \quad (3.6)$$

Kovariantní matice

$$\eta(X_t, \mu, \sum) = \frac{1}{2\pi^{\frac{n}{2}} |\sum|^{\frac{1}{2}}} \cdot e^{-\frac{1}{2} \cdot (X_t - \mu) \sum^{-1} (X_t - \mu)} \quad (3.7)$$

Rovnice Gaussovského rozložení pravděpodobnosti, že bod X náleží v čase t vybrané Gaussové křivce

Na každém novém snímku je pak potřeba porovnat všechny pixely s existujícími Gaussovými rozloženími a je provedena aktualizace jednotlivých vah.

$$\omega_{k,t} = (1 - \alpha) \omega_{k,t-1} + \alpha M_{k,t} \quad (3.8)$$

Kde:

- $\omega_{k,t}$ - váha v čase t ,
- α - učící konstanta,
- $M_{k,t}$ - 0 nebo 1 pokud pixel vyhovoval modelu.

¹MoG - Mixture of Gaussians

Následně je potřeba aktualizovat zbylé hodnoty pro dané rozložení [33].

$$\mu_t = (1 - \rho)\mu_{t-1} + \rho X_t \quad (3.9)$$

$$\sigma_t^2 = (1 - \rho)\sigma_{t-1}^2 + \rho(X_t - \mu_t)^T(X_t - \mu_t) \quad (3.10)$$

$$\rho = \alpha\eta(X_t|\mu_k, \sigma_k) \quad (3.11)$$

Kde:

- μ_t - střední hodnota,
- σ_t^2 - rozptyl.

Všechny křivky jsou řazeny podle poměru své váhy a rozptylu $\frac{\omega}{\sigma}$. Jejich priorita se tedy zvyšuje čím déle je křivka přítomna v modelu pozadí a čím nižší variabilitu má. Pro vyšší výkon se neporovnávají všechny křivky, ale jen omezená skupina křivek B.

$$B = \underset{b}{\operatorname{argmin}} \left(\sum_{k=1}^b \omega_k > T \right) \quad (3.12)$$

Kde:

- T - Minimálnímu množství dat, které jsou zahrnuty do pozadí,
- σ_t^2 - váha jednotlivých křivek.

Pokud je T příliš nízké, je vybrána pouze jedna nejpravděpodobnější distribuce. Naopak pro vysoké T, je model tvořen více Gaussovskými rozloženími a algoritmus je schopný detekovat i velmi malé změny v obraze (např. pohyb listí).



(a)

(b)

Obrázek 3.3: a) Vstupní snímek, b) snímek po odečtení pozadí pomocí MoG.

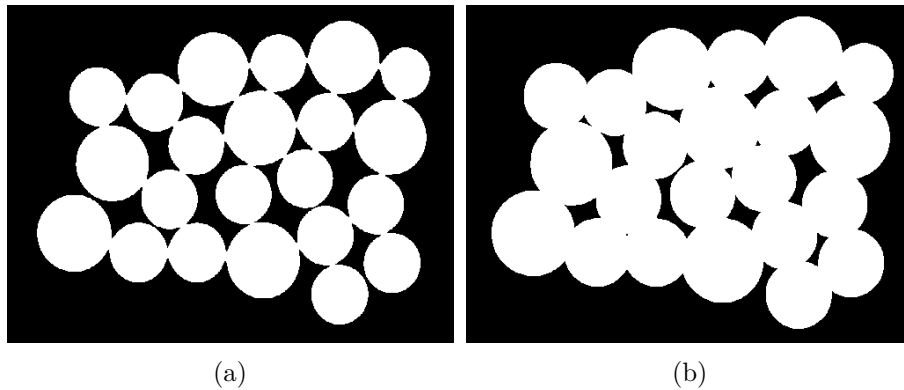
3.1.4 Dilatace

Jedna ze základních operací matematické morfologie je právě dilatace. Operace dilatace se používá pro rozšíření tvarů (vyplnění děr a mezer) ve snímku, a tedy zvětšení těchto tvarů ve vstupním snímku. Pro dilataci objektu ve platí následující rovnice [32].

$$A \oplus B = \bigcup_{b \in B} A_b \quad (3.13)$$

Kde:

- A - binární obrázek,
- B - strukturovací element.



Obrázek 3.4: a) Neupravená maska před aplikací filtru [32], b) Maska po aplikaci dilatačního filtru.

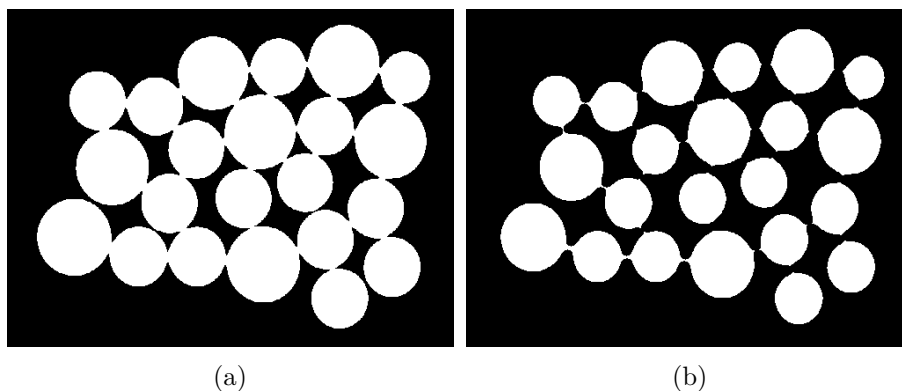
3.1.5 Eroze

Eroze je další ze základních operací matematické morfologie. Základní idea je za pomoci jednoduchého tvaru, jako je například elipsa, prozkoumat vybraný objekt a rozšiřovat veškeré nalezené díry. Tímto může dojít ke zjednodušení tvaru objektu. Pro erozi objektu platí následující rovnice [32].

$$A \ominus B = \bigcap_{b \in B} A_{-b} \quad (3.14)$$

Kde:

- A - binární obrázek,
- B - strukturovací element.



Obrázek 3.5: a) Neupravená maska před aplikací filtrů [32], b) Maska po aplikaci erozního filtru.

3.2 Segmentace obrazu

Počítače a počítačové programy mají velmi omezené možnosti, jaké informace jsou schopny získat z obrazového vstupu bez pokročilé analýzy daného snímku. V případě této práce, požadujeme rozeznání pohybujících se osob.

Při použití naivního postupu se provádí detekce osob po celé ploše snímku, ale jedná se o výkonově a časově velmi náročnou úlohu. Je proto nutné rozlišit pohyb na vybraných snímcích od statického pozadí. Tímto krokem lze získat mnohem menší plochy ze snímku, které je nutné analyzovat na přítomnost postav, což může vést ke znatelnému zrychlení celého procesu detekce a rozpoznání. V této části projdeme některé techniky, jak lokalizovat pohyb v obrazovém snímku a přípravu snímku pro detekci osob.

3.2.1 Odčítání pozadí

Tato metoda, známá také jako detekce popředí, je jedna ze základních metod pro detekci objektů v oblasti počítačového vidění. Pro svou rychlost a jednoduchost jde o jednu z velmi často používaných metod. Metoda odečtu pozadí umožňuje několik přístupů, jak na vstupních snímcích detekovat pohybující se objekty.

Základní myšlenka je pro všechny tyto přístupy podobná - z aktuálního snímku je pixel po pixelu odečteno předem získané pozadí scény. Výsledek toho odečtu je pozitivní maska, které odpovídá pohybujícím se objektům ve scéně. Jelikož v reálném nasazení do detekce vstupují vnější vlivy, jako je měnící se nasvětlení, počasí, ale i komprese snímků, je tato technika sama o sobě špatně použitelná. Pro co nejkvalitnější detekci je tedy nutné před samotným použitím snímku tento snímek upravit.

Tyto úpravy se týkají zejména odstranění šumu ze snímku a malých pohybů (např. listy na stromech). Toho dosahujeme například mírným rozmazáním obrazu, ale tento krok může být na úkor horší detekce. Vliv okolního osvětlení můžeme částečně potlačit převodem snímku z plno-barevného na stupně šedi.

Metoda rozdílu snímků

Díky jednoduchosti algoritmu, popsaného v sekci 3.1.1, a použitým výpočetním operacím, jde o rychlý a paměťově nenáročný algoritmus. Velká nevýhoda tohoto přístupu je právě porovnání pouze dvou snímků. Jelikož mezi těmito dvěma snímky nemusí proběhnout dostatečně výrazný pohyb pro vytvoření kvalitní masky, nebo naopak se může velká část scény změnit (chvilkové zakrytí kamery, skoková změna osvětlení atd.). Výsledná maska pak může být špatně použitelná pro pozdější detekci osob. Z tohoto důvodu je využitelnost metody rozdílu snímků omezená. Nejvíce se hodí pro záznam se statickým pozadím a výrazným pohybem v popředí.

Metoda využívající mediánový filtr

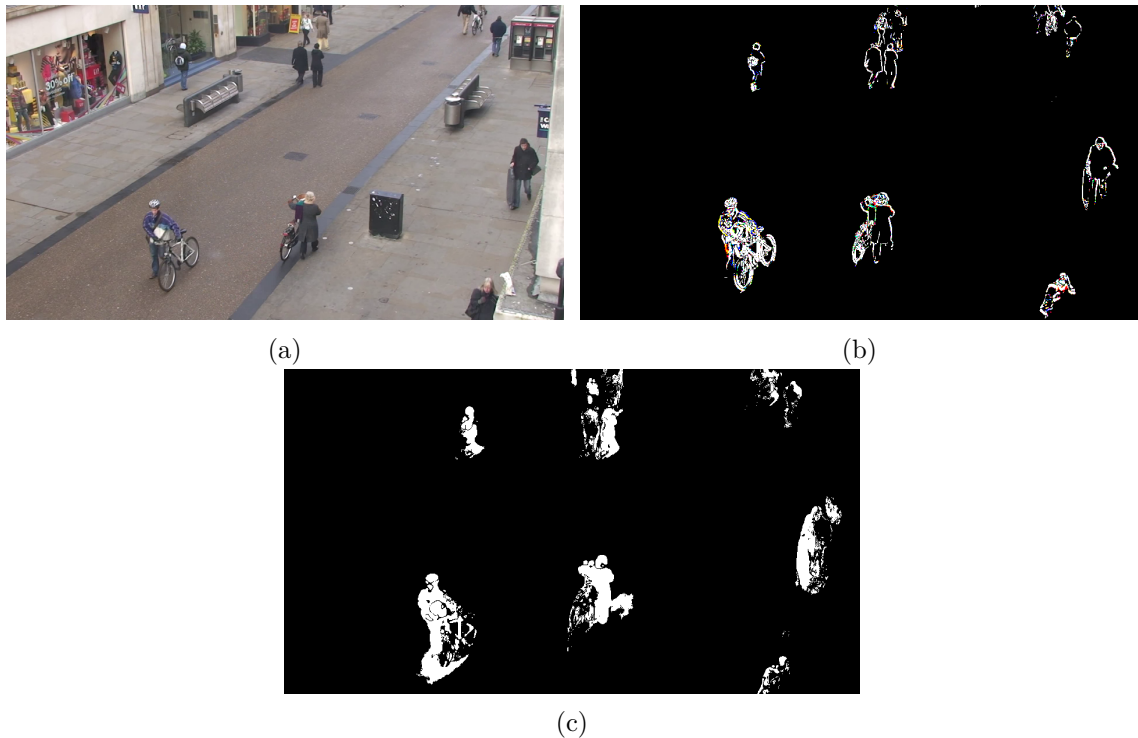
Algoritmus využívající mediánový filtr, viz. 3.1.2, je paměťově i výpočetně náročný, jelikož je nutné uchovávat velké množství snímků v paměti a dochází k neustálému přepočítávání modelu pozadí. Výhodou této metody je ale mnohem lepší výsledná maska než u algoritmu popsaného v sekci 3.1.1. Díky neustálé aktualizaci modelu je výsledná maska vhodná pro další zpracování bez větších úprav, jelikož většina pozadí scény je odfiltrována, obrázek 3.6.

I přes neustálé generování nového modelu, je v určitých případech tato metoda hůře využitelná. Horší výsledky se týkají především případů scén, kde dochází ke změně osvětlení scény, k neustálému rychlému pohybu (např. listy na stromě) nebo naopak k velmi dlouhé změně scény (dlouhodobě zaparkované auto apod.). Tato práce se zabývá sledováním pohybu osob jak ve vnitřních, tak i ve venkovních prostorech, kde dochází k častým změnám osvětlení a množství pohybujících se objektů. Z těchto důvodů je algoritmus využívající mediánový filtr pro detekci pozadí pro tuto práci nevhodný.

Metoda směsice Gaussových křivek

Tento algoritmus popsaný v sekci 3.1.3 si získal obrovskou popularitu za dobu své existence. Za rozšířenost vděčí svým dobrým výsledkům na poli odečítání pozadí. Kvalitní výstup je bohužel vykoupen pomalejším zpracováním vstupních snímků. MoG dokáže pro každý pixel uchovat několik modelů a tím se vypořádat s nechtěnými opakovanými pohyby (např. listy na stromu nebo vlajky na stožárech). Model se také dokáže přizpůsobit změnám osvětlení, které je stále proměnné ve venkovním použití. Díky této přizpůsobivosti se MoG hodí pro takovéto scény.

Za nevýhodu se dá považovat větší množství parametrů, které je potřeba nastavit a ladit. Inicializace je také docela problémová, pokud jsou objekty popředí přítomné na pozadí během inicializace. I pro velmi nízké učící hodnoty α se dostáváme na stovky snímků, než se pozadí scény stane pozadím systému. Přes své nedostatky je metoda založená na směsici Gaussových křivek nejvhodnější pro použití v této práci kvůli kvalitě výstupu, který metoda poskytuje.



Obrázek 3.6: Porovnání metod pro odčítání pozadí a) vstupní obrázek, b) Získaná maska odečtem dvou snímků, c) Maska vytvořená pomocí směsice Gaussových křivek.

3.3 Detekce a popis oblastí zájmu

Výstupem předchozích kroků je binární maska pro každý používaný snímek z videosekvence. Aplikací této masky na původní snímek získáme všechny oblasti, kde byl zaznamenán nějaký pohyb. Díky vnějším vlivům, nedokonalostem použitých algoritmů a nebo zpracováním vstupních dat, jsou reálné objekty často reprezentovány několika binárními objekty (jsou segmentovány).

V případě lidské postavy se často jedná o několik blízkých segmentů - hlava, trup a končetiny. Tato segmentace je způsobena malým pohybem v oblastech některých kloubů a páteře během chůze (zejména ramena a krk), který algoritmy „nevidí“. Je tedy snaha o vhodné sloučení jednotlivých binárních objektů tak, aby následná detekce osob nemusela operovat nad zbytečnými daty. Nad takto získanými oblastmi je již možné provést klasifikaci, zda se na daném místě nachází osoba nebo jiný objekt.

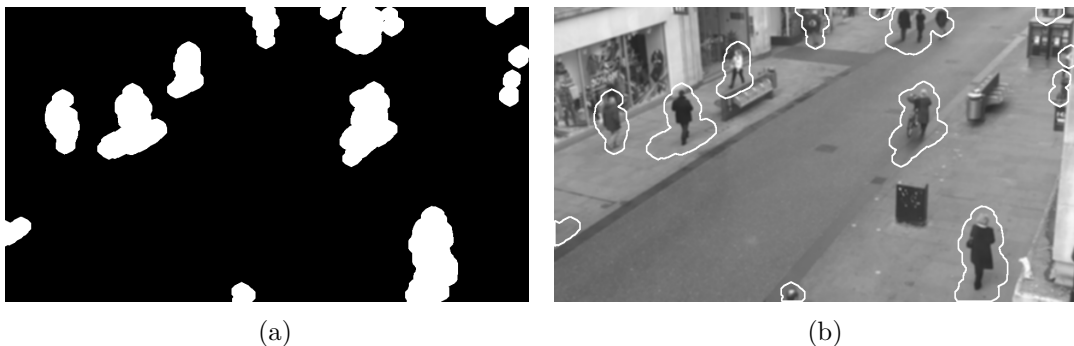
Tyto binární objekty je pak potřeba unikátně popsat pro další práci. Algoritmy pro vyhledávání v obrazových datech se tedy snažíme získat data popisující objekty z vybrané oblasti. Jednotlivé algoritmy se zaměřují na různé charakteristické prvky, které takto získávají ze snímku. Cílem této části je popsat detekci oblastí zájmu, porovnání některých algoritmů pro popis objektů a následně vybrat ten nejvhodnější algoritmus pro následnou klasifikaci.

3.3.1 Detekce binárního objektu

Po aplikaci masky na původní snímek můžeme zjistit, které pixely se změnilly, tedy kde byl vykonán nějaký pohyb. Lze se tedy domnívat, že shluky těchto pixelů, neboli bloby, odpovídají reálným objektům.

Po seskupení pixelů do jednotlivých ploch je možné zpozorovat, že bylo nalezeno větší množství binárních objektů různě fragmentovaných a vnořených do sebe. Pokud připustíme, že hledané osoby mohou být fragmentované mezi více binárních objektů, tak je potřeba tyto fragmenty smysluplně sloučit. Jedním z řešení je nalezení kontur, tedy vyhledání a spojení všech bodů se stejnou intenzitou okolo hranice všech objektů. Touto jednoduchou metodou lze získat o všech oblastech informace jako je velikost nebo centroid.

Zkoumáním jednotlivých vlastností lze pak provést sloučení sobě blízkých objektů u kterých lze předpokládat, že jde o osobu nebo skupinu osob. Na těchto konturách je provedena aproximace pro zjednodušení jejich tvaru pomocí Ramer-Douglas-Peuckerova algoritmu [11][28]. Těmito kroky lze odstranit z dalšího zpracování nevhodné a špatně detekované oblasti, jelikož svými vlastnostmi nemohou odpovídat lidské osobě. Získané binární objekty, lze pak považovat za oblasti zvýšeného zájmu, které obsahují oblasti nad kterými lze provádět další zpracování jako je klasifikace. Pro zjednodušení manipulace se získanými oblastmi každému výslednému binárnímu objektu opíšeme obdélník, který ohraničí vybranou oblast zájmu, ROI².



Obrázek 3.7: Detekce objektů ve snímku a) binární maska, b) detekované oblasti (kontury).

²ROI - Region Of Interest

3.3.2 Scale-Invariant Feature Transform

Zkráceně SIFT, je algoritmus, mimo jiné, pro rozpoznávání objektů publikován již v roce 1999 Davidem Lowem [24]. Tento algoritmus se podobně jako Haarův algoritmus [35] převážně zaměřuje na detekci rohů, jakožto neměnných útvarů s ohledem na rotaci snímku. SIFT přináší zásadní vylepšení, zabývá se škálováním snímku. Přichází snaha, aby každý bod bylo možné najít i ve snímcích s různým měřítkem. Algoritmus se tedy snaží ve snímku nalézt zájmové body a vypočítat jejich deskriptory, kde každý deskriptor má přiřazený zakódovaný popis svého okolí, historiogram gradientů. Ten se snaží zajistit neměnnost datých bodů při transformaci snímku nebo změně nasvícení. Algoritmus lze rozdělit do čtyř základních kroků.

- Detekce lokálních extrémů ve Scale-space.
- Lokalizace klíčových bodů ve snímku.
- Přiřazení orientace klíčovým bodům.
- Vytvoření deskriptorů.

Detekce scale space

Prvním krokem je detekce klíčových bodů, tedy významných bodů, které mohou být opakovaně použity pro identifikaci objektu v různém zobrazení. Klíčové body jsou získány jako minima/maxima rozdílu Gaussových funkcí (DoG³) a jejich konvolucí s původním obrázkem. Tohoto rozdílu dosáhneme odečtením dvou Gaussových funkcí s rozdílnými měřítky.

$$D(x, y, \sigma) = L(x, y, k_i\sigma) - L(x, y, k_j\sigma) \quad (3.15)$$

$$L(x, y, k\sigma) = G(x, y, \sigma) \cdot I(x, y) \quad (3.16)$$

Kde:

$L(x, y, k\sigma)$ - je konvoluce původního snímku $I(x, y)$ s Gaussovým rozmazáním $G(x, y, \sigma)$ na měřítku $k\sigma$.

V momentě vypočítání rozdílu Gaussových funkcí v různých měřítkách, lze lokální minima-maxima považovat za kandidáty na klíčové body. Lokální extrém je získán porovnáním všech jednotlivých pixelů se svými osmi sousedy a devíti sousedy v každé přilehlé vrstvě (měřítku). Ekvivalentem rozdílu Gaussových funkcí je LoG⁴ filtr[23][22]. LoG se ale nepoužívá pro systémy na detekci v reálném čase kvůli své výpočetní náročnosti, naopak pouhý odečet dvou obrázků je výpočetně nenáročný a DoG je tedy používán pro rychlejší detekci.

Lokalizace klíčových bodů

Jakmile jsou pozice potencionálních klíčových bodů známy, tak je potřeba tyto hodnoty upřesnit pro získání lepších výsledků. Jelikož klíčové body jsou lokální extrémy na snímku, tak upřesnění umístění se provede expanzí Scale-space Taylorovou řadou a výsledek je poté podroben prahování.

³DoG - Difference of Gaussians

⁴Laplacian of Gaussian

$$D(x) = D + \frac{\partial D^T}{\partial x} + \frac{1}{2}x^T \frac{\partial^2 D}{\partial x^2} x \quad (3.17)$$

Interpolace lokálních extrémů expanzí Taylorovou řadou. [24]

Tím jsou eliminovány body s nízkým kontrastem. Pro upřesnění výsledných klíčových bodů je také potřeba eliminovat body hran, jelikož DoG je citlivý právě na hrany.

Přiřazení orientace

Pro dosažení invariantnosti klíčových bodů na rotaci snímku, je potřeba každému bodu přiřadit orientaci. Pro okolí každého klíčového bodu je vytvořen histogram gradientů. Histogram je rozdělen na 36 sloupců po 10° , celkem je pokryto celých 360° v okolí vybraného bodu. Každý sloupec obsahuje vážené hodnoty, a podle sloupce s nejvyšší hodnotou se klíčovému bodu přiřadí orientace. Sloupce, které mají aspoň 80% z hodnoty největšího sloupce, se použijí pro vytvoření nového klíčového bodu s danou orientací a shodnými hodnotami jako původní bod.

Vytvoření deskriptorů

Pro každý klíčový bod je potřeba vytvoření deskriptoru, nejčastěji se jedná o histogram gradientů. Pro deskriptor bodu, se vezme okolí o rozměrech 16×16 bodů, které se dále rozdělí na 16 bloků o rozměrech 4×4 bodů. Pro každý takový blok je vytvořen histogram gradientů o 8 sloupcích, celkem se tedy jedná o vektor o 128 sloupcích gradientů pro celé okolí klíčového bodu. Osvětlení, rotace a další transformace snímku mohou mít stále vliv na vybraný bod, proto jsou jednotlivé sloupce deskriptoru váhovány pro zvýšení robustnosti.

Využití a výhody

SIFT algoritmus je nejčastěji používán pro rozpoznávání gest a objektů, vytváření panoramat nebo také 3D modelování. Oproti algoritmům jako je Haarův algoritmus, je velká výhoda malé závislosti na osvětlení díky Scale-space. Naopak zásadní nevýhoda je relativně vysoká výpočetní náročnost, což jej téměř vylučuje z rozsáhlejších sledovacích systémů.

3.3.3 Speede-Up Robust Features

Speede-Up Robust Features neboli SURF[2], je jeden z novějších algoritmů. Své základy má postavené na starším a pomalejším algoritmu SIFT 3.3.2. Algoritmus je velmi podobný se SIFT, liší se ale ve způsobu dosažení svého cíle v jednotlivých krocích. Celý algoritmus může být rozdělen do čtyř částí.

- Detekce klíčových bodů ve snímku.
- Lokalizace klíčových bodů ve snímku.
- Popsání lokalizovaných bodů.
- Párování detekovaných bodů.

Detekce klíčových bodů

Na rozdíl od SIFT se pro aproximaci LoG používá filtrů ve tvaru čtverce místo rozdílů Gaussových funkcí. Výhoda je rychlé filtrování, zvláště pokud je použit integrální obraz definovaný rovnicí 3.18.

$$S(x, y) = \sum_{i=0}^x \sum_{j=0}^y I(i, j) \quad (3.18)$$

Integrální obraz [13].

V integrálním obrazu $S(x, y)$ je možné, v jakémkoli čtvercovém regionu o libovolné velikosti, spočítat sumu pixelů pouze výpočty ve všech čtyřech rozích regionu, což výrazně snižuje výpočetní náročnost. SURF pro vyhledání klíčových bodů ve snímku používá Hessianskou matici. Determinant Hessianské matice je použit jako měřítko lokálních změn okolo vybraného bodu. Použití determinantu není omezeno pouze na zjištění lokálních změn, ale je použit i pro výběr měřítka snímku.

$$H(p, \sigma) = \begin{pmatrix} L_{xx}(p, \sigma) & L_{xy}(p, \sigma) \\ L_{xy}(p, \sigma) & L_{yy}(p, \sigma) \end{pmatrix} \quad (3.19)$$

Hessianova matice pro snímek I v bodě $p = (x, y)$ a měřítkem σ . [2]

Lokalizace klíčových bodů

Jednotlivé klíčové body lze nalézt v různých stupních měřítek snímku. Oproti algoritmu SIFT 3.3.2 se při detekci nepoužívá pyramida pro nalezení jednotlivých klíčových bodů, ale pro získání úrovní SURF používá změnu velikosti Gaussovského jádra. Scale-space je tedy analyzována změnou velikosti filtru raději než změnou rozměrů snímku. Filtr na nejnižší úrovni je čtvercový filtr o velikosti 9x9 pixelů a měřítkem $\sigma = 1.2$, jedná se o aproximaci Gaussovy druhé derivace. Pro lokalizaci klíčových bodů je použité potlačení nemaximálních hodnot v okolí 3x3x3. Jelikož rozdíl měřítek mezi prvními vrstvami každé oktávy je velký, tak je nutné determinant Hessianské matice interpolovat s měřítkem a snímek.

Popis klíčových bodů

Stejně jako u jiných deskriptorů, tak i zde je cílem zajistit unikátní a robustní popis každého klíčového bodu, tak aby byl nezaměnitelný. Popis je tedy získáván pro všechny body nalezené v předchozím kroku. Největší vliv na kvalitu deskriptoru má dimensionalita jednotlivých deskriptorů. Čím více dimenzí, tím lepší vlastnosti pro pozdější párování. Je potřeba, ale vždy najít vhodný kompromis mezi nízkou dimenzí (více false-positive detekcí) a vysokou dimenzí (vysoká přesnost), jelikož je tím přímo ovlivněna komplexnost výpočtů, které je potřeba provést.

Pro dosažení invariance vůči rotaci, je nutné nalézt orientaci klíčového bodu. Toho dosáhneme výpočtem Haarových vlnek ve směru x a y okolo bodu v kruhové oblasti o poloměru $6s$, kde s je vrstva (měřítko), ve které byl bod nalezen. U velkých měřítek získáváme velké vlnky, proto se pro výpočet znovu používají integrální snímky, a tím dochází ke zrychlení výpočtu. Vypočtené vlnky jsou pak váhované Gaussovou funkcí a výsledky jsou reprezentovány jako vektory.

Nakonec je sestrojena čtvercová síť vycentrovaná okolo klíčového bodu a je z ní vyextrahován SURF deskriptor.

Párování

Poslední krok je párování. To je založeno na porovnávání deskriptorů získaných z různých snímků a vyhledání shodných párů.

Využití a výhody

Hlavní výhoda tohoto algoritmu spočívá v rychlejší detekci a popisu oproti algoritmům založených na LoG jako je například SIFT 3.3.2. Rychlost je vykoupena ztrátou přesnosti, ale v publikovaných porovnání[3] je možné zjistit, že nejde o dramatický pokles, a tedy výměna nižší přesnosti za rychlost je výhodná. Algoritmus je vhodný hlavně pro rychlou detekci jednoduchých objektů. Pro tuto práci je použitelný, ale následující algoritmus je vhodnější.

3.3.4 Histogram of Oriented Gradients

Histogram orientovaných gradientů, HOG, je algoritmus pro extrakci příznaků na základě normalizovaném lokálním histogramu orientace gradientu [9]. Základní myšlenka tohoto přístupu předpokládá, že lze popsat vzhled a tvar objektu ve snímku na základě distribuce intenzity gradientů v jednotlivých menších částech. Díky tomu lze dosáhnout lepších výsledků než s předešlými algoritmy. Algoritmus má následující tři kroky.

- Vypočítání gradientu snímku.
- Zjištění orientace bodů v prostoru.
- Seskupení bodů do bloků a normalizace.

Výpočet gradientu

Před samotným zpracováním snímku je vhodné zajistit normalizaci gammy a barevného spektra převedením do stupňů šedé. Výsledné zrychlení dalšího zpracování může být zanedbatelné, ale může se pozitivně projevit. Jako hlavní část tohoto kroku je výpočet gradientu. Ten je získán derivací snímku pomocí vycentrované 1-D masky, která se následně aplikuje konvolucí.

$$D_x = \begin{vmatrix} -1 & 0 & 1 \end{vmatrix} \quad a \quad D_y = \begin{vmatrix} 1 \\ 0 \\ -1 \end{vmatrix} \quad (3.20)$$

Vhodná maska pro detekci osob. [9]

V případě barevného snímku, se pro každou barevnou složku vypočítává gradient zvlášť a ta s největším vektorem je vybrána.

Orientace bodů v prostoru

Druhý krok výpočtu je vytvoření histogramu pro každou buňku. Každý pixel v buňce se váhově podílí na výpočtu orientovaného histogramu na základě vypočteného gradientu v předchozím kroku. Samotný histogram je pak v rozmezí $0^\circ - 180^\circ$ bez použití znaménka nebo $0^\circ - 360^\circ$ s ním. Pro detekci osob nám ale stačí první varianta, která je lehce rychlejší.

Seskupení do bloků a normalizace

Pro z zvýšení robustnosti a přesnosti výpočtu je potřeba snížit vliv osvětlení a kontrastu na snímek. Toho dosáhneme normalizací kontrastu na větší ploše, je tedy potřeba jednotlivé buňky spojit do propojených bloků. Tyto bloky jsou ve vzájemném a částečném překryvu mezi sebou, což znamená, že jednotlivé buňky přispívají do více deskriptorů. Jako optimální parametry se jeví konfigurace 8x8 buněk na blok (16x16 pixelu na blok) s 9 kanály histogramu.

Následně jsou jednotlivé bloky seskupeny do detekčního okna. Proveďte se sečtení histogramů všech bloků v okně a je vytvořen finální 1-D vektor

Využití a výhody

Jedná se o robustní algoritmus pro popis tvaru objektu pomocí struktury gradientů nebo hran. Podobně jako předchozí algoritmy, je stálý, invariantní, vůči změně osvětlení i kontrastu. Je zde i invariantnost vůči transformaci snímku, s jistými omezeními pro rotaci. Hlavní nevýhoda je velká náročnost výpočtu.

Díky vlastnostem a schopnostem tohoto algoritmu, jsem se rozhodl jej použít v této práci spolu s klasifikačním algoritmem popsáním v kapitole 3.4.

3.4 Klasifikace objektů

3.4.1 Adaptive Boosting

AdaBoost [14], patří mezi strojově učící algoritmy, konkrétně do skupiny Boosting algoritmů. Boosting je založen na myšlence využití slabých klasifikátorů pro konstrukci finálního silného klasifikátoru. Každý takový klasifikátor provádí výpočet nad vstupními daty a výstup všech těchto algoritmů je následně zkombinován do váhované sumy, která reprezentuje finální výstup boosteru. S každou iterací, po přidání výsledku slabého klasifikátoru do celkové sumy, se data převáží v závislosti na výsledku právě použitého klasifikátoru. Algoritmus AdaBoost na této myšlence dál staví a špatně klasifikovaným datům zvyšuje váhu D_i , čímž zajišťuje konvergenci. Tím je dosaženo, že klasifikátory v následujících iteracích se budou soustředit na chybně klasifikovaná data z předchozích kroků.

Výpočet vah jednotlivých slabých klasifikátorů algoritmu AdaBoost je rozdělen do tří hlavních kroků a to, výpočet chyby klasifikátoru, výpočet váhy modelu a úprava vah vstupních dat. Jako vstupní podmínky uvažujeme správně ohodnocené trénovací data

$x_i, y_i, x_i \in R^D, y_i \in \{+1; -1\}, i = 1, \dots, m$ a inicializované váhy na $D_1(i) = \frac{1}{m}$.

Jako první, je provedeno trénování slabého klasifikátoru pomocí distribuce D_t s předpokladem $h_t : X \rightarrow \{-1, +1\}$. Následuje výpočet vážené chyby ϵ_t pro špatně klasifikované objekty.

$$\epsilon_t = \sum_i D_t(i) I(y_t \neq h_t(x_i)) \quad (3.21)$$

Poté je vypočítán krok α_t , nebo-li váha klasifikátoru D_t

$$\alpha_t = \frac{1}{2} \ln\left(\frac{1 - \epsilon_t}{1 + \epsilon_t}\right) \quad (3.22)$$

Aktualizace váhy trénovacích bodů D_t .

$$D_{t+1}(t) = \frac{D_t \cdot e^{-\alpha_t y_i h_t(x_i)}}{Z_t} \quad (3.23)$$

Kde:

Z_t - je normalizační konstanta zvolena tak, aby D_{t+1} byla distribucí.

Po dokončení iterací skrz všechny vybrané slabé klasifikátory je předpokládaný výsledek algoritmu Adaboost dán rovnicí 3.24 [14]

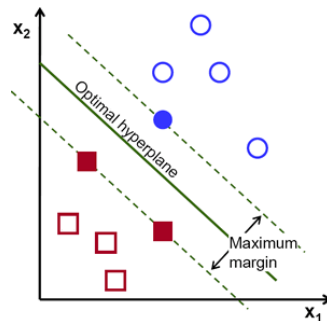
$$H(x) = \text{sign}\left(\sum_{t=1}^T \alpha_t h_t(x)\right) \quad (3.24)$$

Využití a výhody

AdaBoost, díky kombinaci několika dalších učících algoritmů ve svém výpočtu, je v některých případech mnohem méně náchylný k přetrénování, než jiné konkurenční učící algoritmy. Podmínka pro tyto slabé klasifikátory je, že jejich klasifikace musí mít úspěšnost větší než 50%, tedy větší než je pouhé náhodné hádání. Výsledkem pak může být silný klasifikátor. Jedna z nevýhod je pak citlivost na šum a hrany.

3.4.2 Support Vector Machine

Zkráceně SVM, je jeden z algoritmu strojového učení využívajících podpurné vektory [8]. Úlohou SVM je oddělení dat nalezením lineární funkce, nadroviny, která rozdělí trénovací data do patřičných kategorií. Ideální nadrovina je pak taková, která rozdělí data do dvou kategorií s jasnou a co nejširší mezerou mezi dvěma nejbližšími body rozdílných kategorií tak, aby imunita vůči šumu byla co největší. Taková data jsou pak lineárně separovatelná.



Obrázek 3.8: Optimální rozdělení dat do dvou kategorií pomocí nadroviny. [25]

Protože cílem algoritmu je stanovení rozhodovací hranice, není nutné mít k dispozici všechna data. Stačí pouze data, která se nacházejí nejbližší opačné kategorii (v obrázku 3.8 jde o červené čtverce a modrý kruh). Těmto datům říkáme podpurné vektory a roviny, které je protínají jsou podpurné roviny. Takto je poskytnuto dostatečné množství informací pro nalezení rozhodovací hranice.

V případě, že data nejsou lineárně oddělitelná, lze provést nelineární rozdělení pomocí tzv. jádrové transformace prostoru příznaků dat. Lze tak mapovat vstupní data do vyšších dimenzí.

$$f(x) = \omega^T x + b \quad (3.25)$$

Popis lineárního klasifikátoru. [8]

Při použití tohoto klasifikátoru pro jednoduchý klasifikační problém (dvě lineárně separovatelné třídy), pak lze předpokládat pozitivní klasifikaci $y = 1$, pokud $f(x) \geq 0$ nebo negativní klasifikaci $y = -1$ pro $f(x) < 0$.

Pro nalezení separátoru (hlavní nadroviny), který maximalizuje vzdálenosti prvků obou tříd, je potřeba minimalizovat funkce pomocí váhových koeficientů [8].

$$f(x) = \text{sign}([\sum_{i=1}^n \alpha_i y_i k(\vec{x}_i, \vec{z})] + b) \quad (3.26)$$

Rozsepsaný klasifikátor s váhami pro jednotlivé prvky z trénovací množiny. [8]

$$\min \frac{1}{2} \sum_{i,j=1}^m \alpha_i \alpha_j y_i y_j (x_i \cdot x_j) \quad (3.27)$$

Kde:

- α_i - váhový koeficient,
- y_i - klasifikace vstupu $y_i \{-1, +1\}$,
- x_i - vstup.

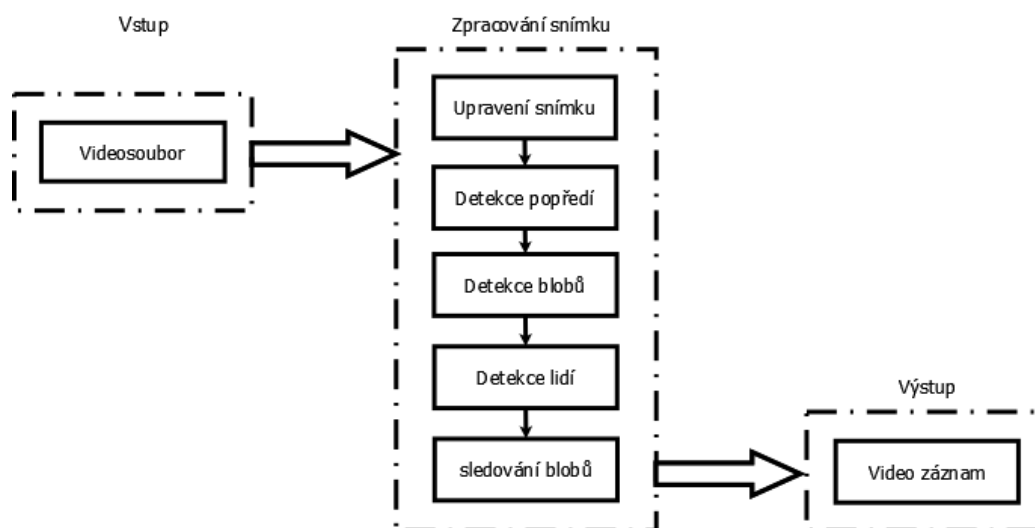
Využití a výhody

SVM klasifikátor se především vyznačuje svou rychlostí a přesností oproti jiným klasifikátorům v rámci klasifikace osob, často ale záleží na konkrétním použití, a je tedy téměř nemožné tento výrok zobecnit na všechny použití. Pro tuto práci jsem se rozhodl použít právě SVM klasifikátor spolu s HOG deskriptorem popsáním v sekci 3.3.4. Výpočetně nejnáročnější část tohoto algoritmu je pak extrakce HOG příznaků. Více informací bude probráno v následující kapitole.

Kapitola 4

Návrh a implementace aplikace

Tato část práce, se zabývá návrhem a implementací aplikace pro sledování osob ze stacionární kamery ve venkovním prostředí. V aplikaci je nutné zohlednit, právě venkovní využití celého systému, je tedy potřeba brát tento požadavek na zřetel během celého vývoje. Velký důraz, během návrhu a následující implementace, musí být kladen i na rychlost zpracování jednotlivých snímků, tak aby celá operace mohla probíhat co nejlépe reálnému času. Obrázek 4.1 zobrazuje zjednodušený model aplikace s vyznačenými hlavními prvky. Návrh aplikace se opírá o teorii popsanou v předchozích kapitolách.



Obrázek 4.1: Návrh aplikace

4.1 Použité nástroje

Pro tuto práci jsem zvolil programovací jazyk C++ spolu s knihovnou OpenCV (ver. 3.1.0). Použitá knihovna OpenCV je právě zaměřena na rozpoznávání obrazu, a byla tedy vhodnou volbou. Jazyk C++ jsem zvolil díky jeho rychlosti v daných úlohách v porovnání s často používanými jazyky Python nebo Java. Pro vývoj bylo zvoleno vývojové prostředí Code::Blocks spolu s kompilátorem GCC. Implementace této práce nevyužívá možnosti grafického procesoru (GPU), jelikož v OpenCV dochází k omezení na jediného výrobce. Celá aplikace je díky použití zvolených nástrojů platformě nezávislá, ale celý vývoj i testování proběhlo pouze pod operačním systémem Ubuntu 14.04.

4.2 Vstupní data

V první části programu uživatel volí jeden z hlavních vstupních parametrů, zdroj vstupního videa. Aplikace umožňuje zvolit libovolný videozáznam uložený na disku nebo přímý datový výstup ze statické videokamery připojené k PC. Nezávisle na zvoleném zdroji, se pro další zpracování získávají jednotlivé snímky v nekonečném cyklu. Jelikož čtení dat ze zdroje je blokující operace, tak je celá tato činnost přesunuta do vlastního vlákna. Tím je zajištěno, že samotná práce s daty nemusí neustále čekat na přečtení dat z videosouboru nebo kamery. Pro vývoj a testování aplikace je používán právě videozáznam, který umožňuje rychlý vývoj.

4.3 Zpracování snímku

Jedná se o nejdůležitější část programu, která zajišťuje celou logiku zpracování jednotlivých snímků, následnou detekci osob a jejich sledování.

4.3.1 Úprava snímku

Jak bylo popsáno v kapitole 3, vstupní snímek je potřeba upravit tak, aby následné zpracování mohlo proběhnout rychleji a detekce chodců byla kvalitnější. Na zvýšení rychlosti má v této části největší vliv barevná hloubka a fyzické rozměry snímku. O tyto transformace se stará funkce `changeColorSize`.

Nejdříve je zmenšen snímek na polovinu, pokud rozměry snímku přesahují 1000x600 pixelů. Tato maximální hodnota byla vybrána jako ideální v poměru velikost snímku/rychlost zpracování. Převodem RGB barevné palety do odstínů šedé neztrácíme žádné podstatné informace pro detekci pohybu a pohybujících se objektů. Snímek je tedy převeden do odstínů šedé podle standardu BT.601 4.1.

$$I = 0.299R + 0.587G + 0.114B \quad (4.1)$$

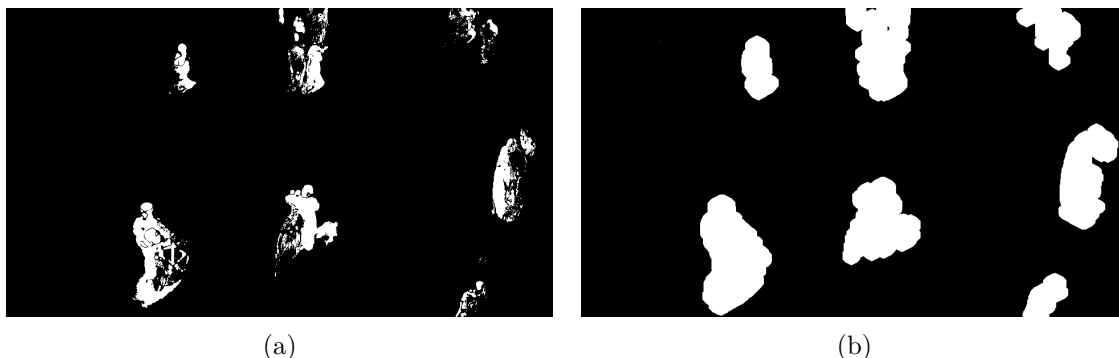
Rovnice pro převod RGB snímku do odstínů šedé. [19]

Kromě omezení barevné hloubky, a tím nutnosti zpracovávat méně dat, lze tímto převodem zmírnit vliv osvětlení scény na výsledek detekce. V rámci zlepšení tvorby binární masky při detekci popředí, je vstupní snímek několikanásobně rozmazán Gaussovým rozostřením. Tímto krokem lze odstranit šum z obrazu, ale i nechtěný drobný pohyb jako je vlnící listí na stromech nebo drobné pohybující se předměty.

4.3.2 Detekce objektů v popředí

Po úpravách, popsanych v části 4.3.1, je možné začít od snímku odečítat pozadí scény resp. detekovat popředí. Při použití vstupního videa se snímkovou frekvencí 24 snímků za vteřinu a větší, nejsou změny v pohybu chodců mezi jednotlivými snímky nijak výrazné a nemělo by tedy docházet k degradaci detekce a následovného sledování při vynechání několika snímků. Zároveň je nevhodné vynechat příliš mnoho snímků, protože následná detekce a sledování osob může značně utrpět, což bylo potvrzeno experimenty při návrhu algoritmu pro detekci osob. Z tohoto důvodu, jsem se rozhodl zpracovávat každý snímek i za cenu větší výpočetní zátěže.

Pro detekci popředí jsem zvolil metodu založenou na směsici Gaussových křivek, popsanou v 3.2.1. Detekce popředí probíhá ve funkci `getForeground`. I přes předzpracování snímku, popsaném v 4.3.1, maska vytvořená odečtením pozadí často obsahuje spousty defektů. Pro odstranění velmi malých i větších defektů (zbylý šum, nevhodné nasvětlení způsobující degradaci snímku v jeho částech aj.), jsou na masku několikanásobně aplikovány morfologické filtry dilatace a eroze, popsané v 3.1.4 a 3.1.5. Jako nejvhodnější tvary zrna pro tyto morfologické operace se osvědčily elipsa a obdelník. Takto upravená binární maska pak umožňuje v daném snímku vyhledání všech spojitých oblastí bílých pixelů, kontur neboli v tomto případě blobů.



Obrázek 4.2: a) Detekované popředí po aplikaci MoG, b) výstupní maska po aplikaci morfologických filtrů.

4.3.3 Detekce blobů

Upravená maska z předchozí části již umožňuje detekci jednotlivých blobů. V každém zpracovávaném snímku se pomocí binární masky vyhledávají všechny spojitě oblasti bílých pixelů - kontury neboli bloby. Občas se může stát, že nalezené bloby nepokrývají celý hledaný objekt. Pro lepší klasifikaci těchto objektů je tedy vhodné uvažovat i bezprostřední okolí daného blobu, ROI (oblast zájmu).

Opsáním obdelníku daného blobu, získáme právě i bezprostřední okolí, které může obsahovat části osob. V případě většího počtu nalezených blobů na snímku, může docházet k překryvům jednotlivých obdelníků. Přestože nám jde hlavně o zmenšení plochy, kterou je potřeba prozkoumat během detekce, tak je možné tyto oblasti zájmu sloučit. Můžeme se tak vyhnout, aby jednotlivé části osob, zejména končetiny a hlava, byly brány jako několik samostatných objektů v případě chybné detekce popředí. Výsledné binární objekty, jsou reprezentovány středem svého opsaného obdelníku, jeho obsahem a souřadnicemi. Všechny tyto vlastnosti jsou uchovávány pro pozdější použití.

4.3.4 Detekce osob

Získané oblasti zájmu je již možné prohledávat na přítomnost chodců. Tato část má dva kroky, které mají kritický význam pro generování kvalitních dat pro následné sledování. Vše začíná volbou vhodného algoritmu pro popis obrazových dat. Po porovnání několika přístupů jsem vybral algoritmus HOG, více v kapitole 3.3.4.

Histogram orientovaných gradientů, tak jak je implementován v použité knihovně OpenCV, má několik variant a vstupních parametrů, které mají zásadní vliv na rychlost zpracování a kvalitu popisu dat. Varianta použitá v této práci je postavena na metodě **HOGDescriptor::detectMultiScale**, která provádí detekci pomocí okna ve více měřítcích.

Největší vliv na detekci mají dva parametry. Jedná se o *winStride*, neboli velikost okna, pomocí kterého se posunem prohledává daná oblast. Po experimentech se jako nejvýhodnější velikost okna jeví 8x8 pixelů. Menší okno pak vede k pomalému zpracování a naopak při použití větší velikosti se ztrácí přesnost. Druhý (volitelný) parametr je měřítko, podle kterého je tvořena pyramida snímku. Každá vrstva pyramidy je zmenšena a případně i vyhlazena. Tento parametr kontroluje, jak bude daný snímek zmenšen, a tedy kolik stupňů pyramidy bude potřeba postavit. Obecně se tato hodnota pohybuje v intervalu $< 1.01; 1.5 >$, kdy menší hodnota znamená větší počet vrstev pyramidy a delší čas pro zpracování. Experimentálně hodnota měřítko byla zvolena 1.09.

Důležitý krok, který má největší vliv na sledování, je klasifikace detekovaných objektů. Pro každý objekt se na základě svého popisu rozhodne, jestli se jedná o chodce nebo ne. Pro tento krok jsem vybral rychlý a spolehlivý algoritmus strojového učení SVM 3.4.2. Pro klasifikaci vybraných objektů byla použita předtrénovaná data v knihovně OpenCV pro detekci osob.

Výstupem detekce osob je seznam lokací v aktuálním snímku, kde se nachází chodci.

4.3.5 Sledování osob

V této části dochází k samotnému sledování detekovaných chodců mezi jednotlivými snímky. V oblasti rozpoznávání obrazu existuje mnoho postupů a návrhů pro vyřešení velmi složitého úkolu, a to detekované objekty (v případě této práce se jedná o chodce) úspěšně sledovat po celou dobu přítomnosti ve sledované oblasti. Jedny z těchto algoritmů jsou Mean Shift [15], Kalmanův filtr [21], částicový filtr a další. Všechny algoritmy, zabývající se sledováním objektů, musí překonat stejné problémy. Obecně se dá říci, že každý z těchto algoritmů má své nedostatky a použitím se hodí na některé úlohy více než jiné.

Jeden z nejzákladnějších problémů, se kterým se lze setkat, je překrytí sledovaných osob. V tomto případě se algoritmus potřebuje vypořádat s faktem, že tam kde byly původně osoby dvě, je najednou jediná a v následujícím okamžiku jsou znovu dvě. V moment překrytí se ztrácí přehled o daných osobách a po jejich „rozpojení“ je nutné korektně určit, o které osoby jde, i když během překrytí změnila třeba směr chůze. V případě většího počtu osob nebo davu na ulici se komplexnost řešení rychle zvyšuje. Podobná situace nastává, pokud se sledovaná osoba „ztrácí“ z dohledu. Může jít pouze o chybu klasifikátoru, který nebyl schopen po určitou dobu danou osobu klasifikovat jako člověka nebo i o prosté zakrytí překážkou (projíždějící autobus, stánek apod.). Obě tyto situace by měl dobrý sledovací algoritmus zvládnout. Často se tak děje za použití predikce pohybu a následných korekcí.

V této práci nebyl použit žádný z dostupných sledovacích algoritmů, ale byl navrhnout nový, tak aby co nejlépe řešil výše popsání situace. Cílem bylo vytvořit algoritmus, který dokáže v obrazovém záznamu sledovat několik osob ve venkovním prostředí s ohledem na

rychlost zpracování a kvalitu výstupu. Byl tedy zvolen postup porovnávání barevných složek osoby a vzdálenosti mezi jednotlivými snímky.

Při návrhu algoritmu bylo potřeba vymyslet způsob, jakým si uchovávat informace o jednotlivých postavách tak, aby bylo možné mezi nimi hledat stejné nebo velmi podobné páry. Pro tento účel byl vytvořen vektor *PeopleArray* založený na třídě *myMatrix*. V této struktuře jsou uchovány všechny informace o každé detekované osobě. Tyto informace se neomezují jen na pozici osoby, ale i její histogramy, samotný snímek osoby, cesta kudy osoba prošla (trajektorii) a další. Jsou zde uloženy i záznamy o dané osobě z jednoho následujícího snímku, pokud je ještě někdy detekována.

Téměř celý navržený algoritmus pro sledování se nachází ve funkci **trackHumans**, která jako vstupní parametry má detekované osoby (více v 4.3.4) a vektor *PeopleArray*. Výstup algoritmu jsou pak všechny sledované osoby a informace nově získané z aktuálního snímku.

Algoritmus 1 Pseudokód navrhovaného algoritmu pro sledování chodců

Vstup:

locationHuman: Vektor osob detekovaných v aktuálním snímku

PeopleArray: Vektor všech osob z předešlých snímků

Výstup:

PeopleArray: Aktualizovaný vektor všech detekovaných osob

```
1: if locationHuman or PeopleArray jsou prázdné then
2:   for i < locationHuman do
3:     | Ulož postavu z locationHuman do PeopleArray
4:   end for
5: else
6:   for j < locationHuman do
7:     | Vlož j-tý prvek do proměnné a
8:     | Vypočti histogramy barevných složek pro a
9:     for i < PeopleArray do
10:      | Vlož i-tý prvek do proměnné b
11:      | if prvek b nemá uložený histogram then
12:        | Vypočti histogramy barevných složek pro b
13:      | end if
14:      | Vypočti fyzickou vzdálenost  $Len = a - b$ 
15:      | Vypočti Eukleidovskou vzdálenost euclid mezi a a b
16:      | if  $Len \leq 2\%$  celkové plochy snímku then
17:        | if  $euclid \leq euclid$  předchozí porovnávané postavy then
18:          | Uchovej i-tý a j-tý prvek jako potencionální shodu
19:        | end if
20:      | else if  $Len > 2\%$  and  $Len \leq 4\%$  celkové plochy snímku then
21:        | if  $euclid \leq euclid$  předchozí porovnávané postavy then
22:          | Uchovej i-tý a j-tý prvek jako potencionální shodu
23:        | else
24:          | i-tý prvek a nemá žádnou shodu, je to nová osoba
25:        | end if
26:      | else
27:        | i-tý prvek a nemá žádnou shodu, je to nová osoba
28:      | end if
29:    | end for
30:  | end for
31: end if
```

Základní část algoritmu pro sledování chodců se skládá z jediné rozvětvené *if* podmínky. Je provedena kontrola, zda již byly detekovány nějaké osoby v předchozích snímcích. V případě, že žádnou osobu nesledujeme nebo se dlouho žádná osoba neobjevila v záznamu, tak jsou všechny aktuální postavy z vektoru *locationHuman* přesunuty do *PeopleArray*. V opačné situaci máme z předchozích snímků a z aktuálně zpracovávaného snímku detekované chodce. Nejjednodušší způsob pro spárování jednotlivých chodců je porovnání všech detekovaných osob z aktuálního snímku oproti všem dosavadním postavám. Pro každou postavu z *locationHuman* se získají histogramy jednotlivých barevných RGB složek a vypočte se střed oblasti zájmu kolem daného chodce. Následně dochází k zanoření a porovnávání s jednotlivými osobami z *PeopleArray*. Zde je nejprve provedena kontrola, zda porovnávaná osoba má svůj histogram, a ten je případně získán. Stejně jako v předchozím kroku, je vypočítán střed oblasti zájmu kolem chodce, a následně je volána funkce **calckHistogram_feature**.

Funkce **calckHistogram_feature** se stará o výpočet Eukleidovské vzdálenosti dvou postav. V případě této práce je Eukleidovská vzdálenost 4.2 použita jako míra podobnosti, kdy hodnoty blízkí se nule znamenají větší shodu. V případě identické osoby a stejné pozici ve snímku je vzdálenost 0.0 a s měnícími se vlastnosti (pozice a barevné složení osoby) tato hodnota stoupá.

$$R_{i,j} = \sqrt{\left(\frac{\Delta_x}{X_{img}}\right)^2 + \left(\frac{\Delta_y}{X_{img}}\right)^2 + \left(\frac{\Delta_B}{255}\right)^2 + \left(\frac{\Delta_G}{255}\right)^2 + \left(\frac{\Delta_R}{255}\right)^2} \quad (4.2)$$

Upravená rovnice použitá pro výpočet Eukleidovská vzdálenosti. [10]

Kde:

- X_{img}, X_{img} - Celkové rozměry vstupního snímku.
- Δ_x, Δ_y - Rozdíl středů osob ve snímku I_t a I_{t-n} na souřadnicích x a y.
- $\Delta_B, \Delta_G, \Delta_R$ - Rozdíl histogramů jednotlivých barevných složek vybraných osob.

Může nastat situace, kdy jsou dva chodci oblečeni do oděvů v podobných barvách a nacházejí se v těsné blízkosti u sebe. V tomto případě budou rozdíly histogramů barevných složek velmi malé. Fyzická vzdálenost mezi takovými osobami je také velmi malá, ale je možné vynásobení eukleidovská vzdálenosti středů obou osob konstantou. Výsledná vzdálenost blízké osoby se tak zvýší, ale vyhledávaná osoba si uchová nejmenší vzdálenost. Jedná se o největší limitaci, kdy za určitých podmínek dochází k zmatení algoritmu.

Následně probíhá třídění na základě fyzických vzdáleností jednotlivých osob a získaných eukleidovských vzdáleností. Osoby se třídí podle třech podmínek.

- Fyzická vzdálenost objektů je menší, než poloměr kružnice představující 2% celkové velikosti vstupního snímku krát vstupní parametr *shortSearch*.
- Fyzická vzdálenost je v rozmezí 2% až 4% velikosti vstupního snímku.
- Fyzická vzdálenost je větší, než 4% velikosti vstupního snímku krát vstupní parametr *longSearch*

Nakonec je podle daných podmínek vybrána nejlepší shoda mezi osobou z aktuálního snímku a jednou z osob sledovaných v předchozích krocích. V případě, že nelze nalézt dostatečnou podobnost, je tato osoba považována za novou.

Takto je teoreticky zajištěné dostatečně kvalitní rozpoznání konkrétního chodce mezi snímky. Překrytí i dočasná ztráta sledované osoby by pak mělo být pro tento algoritmus zvládnutelné.

4.3.6 Záznam trajektorie sledované osoby

Poslední aktivní částí aplikace je zaznamenání a vykreslení zjištěné trajektorie sledovaných chodců.

Celá logika zaznamenávání a ukládání trajektorií je obsažena ve funkci **printTrace**. Každá detekovaná osoba dostává do začátku 25 bodů. Tyto body se po jednom odečítají vždy, když daná osoba není detekována v aktuálním snímku.

Podobně jako při sledování chodců, je potřeba projít celý seznam všech sledovaných osob. Každá osoba je zkontrolována na počet bodů, které vlastní. V momentě, kdy se počet bodů dostane na nulu, je daná osoba odstraněna ze seznamu. Tímto jsem zajistil, že seznam sledovaných chodců nezačne nekontrolovatelně růst a výpočetní náročnost procházení seznamu chodců zůstane na podobně úrovni po většinu času běhu aplikace. Před samotným odstraněním osoby, je její trajektorie uložena do snímku pro odstraněné chodce *PeopleGone*. Následně je odstraněna i z výstupního snímku.

Pokud má osoba stále dostatek bodů, provede se kontrola na přítomnost nalezeného páru z předchozího kroku v 4.3.5. V kladném případě se uloží bod ze středů zájmových oblastí dvojice do vektoru s trajektorií pro pozdější vykreslování. Následně se projde celá trajektorie a každý bod se vykreslí a spojí. Barva vykreslené trajektorie je pro každého sledovaného chodce unikátní a uchovává se po celou dobu jeho sledování. Poté jsou původní informace o chodci nahrazeny aktuálními. V případě, že dotazovaná osoba nebyla v aktuálním snímku spárována, tak je odečten jeden z bodů a přechází se na další osobu.

4.4 Výstup

Poslední vstupní parametr této aplikace je volitelný. V případě nepoužití tohoto parametru, je k dispozici pouze výstup na obrazovku bez možnosti uchování dat. Naopak při zadání validní cesty a dostupnosti vhodných video kodeků, je výstup ukládán do zadaného souboru, spolu se zobrazením na obrazovce.

Pro přehlednost výstupu je možné všechny vykreslené trajektorie z výstupu odstranit klávesou „c“, a naopak pro vykreslení všech trajektorií slouží klávesa „t“.

Kapitola 5

Testování

V této části se zaměříme na testování navrženého algoritmu na několika videozáznamech a ověření funkčnosti celého systému. Rozebereme zde výhody a nevýhody navrženého algoritmu. Zaměříme se i na rychlost zpracování vstupních dat.

Všechny záznamy byly pořízené ze statických kamer z různých úhlů a vzdáleností od sledované scény. Pro snížení výpočetní náročnosti jsou všechny vstupní záznamy přesahující rozměry 1000x600 pixelů zmenšeny na polovinu. Jeden z testovacích záznamů zachycuje reálný pohyb chodců po ulici. Ostatní videozáznamy jsou navrženy pro otestování různých scénářů. U jednotlivých záznamů představím různé situace, které nastaly při sledování spolu s vysvětlujícím komentářem.

5.1 Sledování chodců v reálném prostředí

Pro tento test bylo vybráno video [29], které zachycuje velké množství chodců v reálném prostředí. Jedná se o velmi složité video pro sledování chodců kvůli velkému množství pohybu a nižší kvalitě videozáznamu. Dochází zde ke křížení jednotlivých osob i skupin chodců mezi sebou. Několik osob vstupuje do záběru v jednom bodě a poté dochází k jejich odpojení. V této práci navržený sledovací algoritmus, je založený na histogramech barevných kanálů. Použité video obsahuje velké množství osob v tmavých kabátech a tmavých nebo modrých kalhotách. Jsou zde tedy velmi těžké podmínky pro sledování založené na barvách.

5.1.1 Detekce a klasifikace



Obrázek 5.1: a) detekce objektů, b) klasifikace chodců.

Základem sledování je dobrá detekce a klasifikace sledovaných objektů, v našem případě chodců. Jak lze vidět na obrázku 5.1a, detekce chodců proběhla v pořádku, ale klasifikace nebyla schopná ve všech případech správně určit, že se jedná o lidské postavy 5.1b. Příčin zde může být několik, ale jako hlavní příčinu bych označil nevhodně natrénovaný klasifikátor SVM. Klasifikace také neproběhla v pořádku v místech spodního a horního okraje záznamu. Ve spodním okraji se často nacházejí chodci, kteří nejsou v záznamu celou postavou, ale jen od pasu nahoru. V takových případech klasifikátor natrénovaný na celou postavu selhává. V případě horního okraje jsou chodci naopak velmi malí, a ani jejich zvětšení na velikost potřebnou pro klasifikátor nepřináší zlepšení výsledků. Použitý SVM klasifikátor byl natrénován pouze na lidskou postavu. V použitém záznamu se vyskytuje několik osob jdoucích s kolem nebo kočárkem, tyto osoby jsou tedy neklasifikovány.

5.1.2 Sledování chodců



Obrázek 5.2: a) chodci vstoupili do scény blízko sebe, b) skupina podobně oblečených chodců.

V případě kdy několik osob vstoupí do záznamu společně a jsou blízko sebe, může docházet k nesprávným klasifikacím. Objekt může být označen za jednoho chodce i přesto, že jde ve skutečnosti o dva nebo více chodců. V tento moment, sledujeme jednu „osobu“, ale v momentě kdy se rozdělí, tak dochází ke zmatení sledovacího algoritmu. Algoritmus neví, kterou z osob sledovat. Dochází tedy k přepínání trajektorie mezi těmito chodci až do doby, než se od sebe dostatečně vzdálí a jedna osoba (nebo více) je detekována jako nová.

Podobný případ nastává u stejně oděné skupiny chodců. Jsou velmi blízko u sebe a mají téměř stejné histogramy. Dochází tedy také k přepínání trajektorií. Je vhodné si také povšimnout, že výloha, která se nachází na levé straně záběru odráží dění na ulici. Může dojít ke zmatení detektoru a sledovacího algoritmu, odrazy postav ve výloze jsou pak počítány jako reálné postavy.

5.1.3 Zhodnocení

Jedná se o složité video pro testování. Po nalezení vhodných hodnot, které pomáhají s rozřazením osob během sledování, jsou ale výstupy uspokojivé. Dochází k občasnému přepínání stop mezi chodci, ale ve většině případů se tak děje mezi chodci jdoucími stejným směrem.

5.2 Sledování chodců v kontrolovaném prostředí 1

Jedná se o několik krátkých videí, simulující různé scénáře. Kamera je umístěna v prvním poschodí budovy za sklem a snímá silnici s lehkým provozem aut a cyklistů. Záznam je natočen v pozdních odpoledních hodinách, při velmi silném svitu zapadajícího slunce. Sledované osoby jsou v různobarevných oděvech.

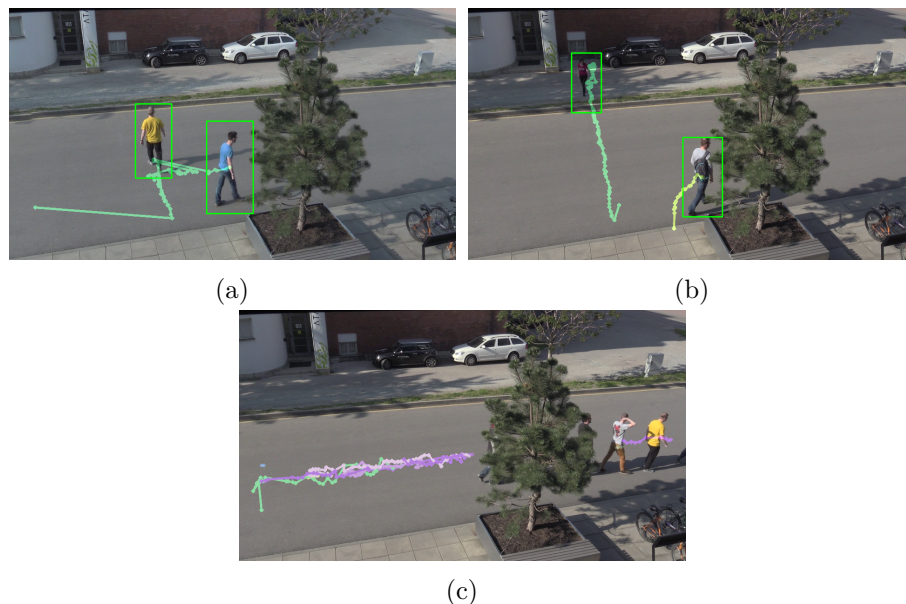
5.2.1 Detekce a klasifikace



Obrázek 5.3: a) detekce objektů, b) klasifikace chodců.

Kvůli nepříznivým světelným podmínkám, jsou tvořeny kolem chodců velmi dlouhé stíny ve tvaru lidské postavy. I přes tuto překážku, neměl klasifikátor s tímto záznamem závažné problémy, ale podobně jako u záznamu popsaném v 5.1 často dochází k horší klasifikaci chodců v horní části snímku.

5.2.2 Sledování chodců



Obrázek 5.4: a) Protnutí chodců, b) samostatný chodec, c) několik chodců jdoucích za sebou.

Při scéně zobrazené v 5.4a dochází k opožděné detekci jednoho z chodců (modré oblečení). Sledování tedy začíná v momentě, kdy jsou obě osoby velmi blízko sebe a částečně se překrývají. V tento moment jsou vnímány jako jedna osoba a nastává stejná situace jako v 5.2a. V okamžik, kdy se od sebe oddělí, dochází ke zmatení sledovacího algoritmu a přepínání trajektorie daných osob.

Na této scéně 5.4b lze pozorovat, že sledování jedné osoby probíhá korektně, a to i v horní části záznamu.

V poslední scéně 5.4c můžeme pozorovat, že i zde dochází k přepínání trajektorií mezi chodci. Kvůli umístění kamery, bylo nutné zvětšit okruh ve kterém se vyhledává osoba do páru. Kvůli této úpravě parametrů, osoby jdoucí blízko sebe se mohou často párovat mezi sebou, a dochází tak ke zhoršenému sledování jednotlivých osob.

5.2.3 Zhodnocení

Jedná se o záznamy natáčené přímo pro testování této práce. Podařilo se ověřit, že sledování jedné osoby navržený systém zvládá bez problému, a to na většině plochy snímku. Stejně jako v testu 5.1, dochází k problémům se sledováním, pokud jsou osoby poprvé ve scéně detekovány velmi blízko sebe. Záznamy obsahují několik cyklistů, kteří jsou správně nedetekováni, i pár jedoucích vozidel, která zakrývají sledované osoby.

V těchto záznamech lze také pozorovat několik značných propadů v rychlosti zpracování. Tyto propady jsou způsobeny nepříznivým osvětlením a automatickým ostřením použité kamery. K propadům dochází právě kvůli větším změnám v obraze, které donutí navržený algoritmus provést detekci osob na celém snímku, místo pouze vybraných oblastí.

5.3 Klasifikace a rychlost sledování

Z důvodu zaměření této práce na sledování osob, tak je potřeba dosáhnout co nejrychlejšího zpracování, detekce a následného sledování osob, ideálně v reálném čase. Na tento požadavek byl brán ohled po dobu vývoje tohoto algoritmu. Všechny jednotlivé části aplikace mají přímý vliv na rychlost zpracování každého snímku. Je ale vždy potřeba udělat kompromis mezi rychlostí výpočtů a kvalitou výstupu. V dnešní době „levného výkonu“, není nutné se již omezovat rychlostí. Z tohoto důvodu jsem využil například MoG pro detekci popředí. Jedná se o pomalejší metodu oproti jiným [5], ale o to kvalitnější výstup poskytuje. Ze stejného důvodu byl zvolen i Histogram orientovaných gradientů pro detekci osob. Právě algoritmus HoG 3.3.4 se projevil jako výrazné úzké místo pro výkon. Bylo potřeba zvolit vhodné parametry, a měřítko pro sestavení pyramid se ukázalo jako nejnáročnější na výkon. Byla použita hodnota 1.09 jako kompromis mezi dostatečnou detekcí a rychlostí. Hodnoty např. větší než 1.3 způsobovaly, že algoritmus nebyl schopen detekovat téměř žádné osoby a naopak hodnoty menší než 1.09 znamenaly obrovskou výpočetní zátěž bez výrazněji lepších výsledků.

V této práci byl použit předtrénovaný klasifikátor z knihovny OpenCV, který nebyl dále upravován. Pro klasifikátor byl vypočten parametr *true positive* (TP), který vyjadřuje počet správně identifikovaných osob.

$$TP = \frac{\text{správně identifikované osoby}}{\text{celkový počet pozitivních objektů}} \quad (5.1)$$

Vzorec na výpočet parametru *true positive*

Pro zjištění tohoto parametru, bylo vygenerováno deset snímků z videozáznamu. Následně pro každý takový snímek proběhlo zjištění počtu detekovaných objektů, počtu osob a počtu objektů klasifikovaných jako lidská postava. Hodnoty ze všech snímků byly akumulovány a následně byl vypočítán parametr TP 5.1. Zjištěná hodnota byla 0.7, tedy 70% úspěšnost. Na výslednou hodnotu TP, mohl mít negativní dopad použitý videozáznam. Použitá scéna byla nasvícena ostrým slunečním světlem.

Měření rychlosti zpracování videosouborů bylo provedeno na několika videozáznamech o různých délkách 5.1. Na jednotlivých záznamech se vyskytuje různé množství sledovaných osob, což má přímý vliv na rychlost zpracování každého jednotlivého snímku. Měření rychlosti proběhlo na finální verzi aplikace.

	Town	Ulice1	Ulice2	Ulice3	Ulice4	Ulice5
Rozlišení	960x540	960x540	960x540	960x540	960x540	960x540
Kodek	XviD	H.264	H.264	H.264	H.264	H.264
Snímková frekvence [FPS]	25	25	25	25	25	25
Délka záznamu [s]	300	104	15	50	77	61
Zpracování snímku [s]	0.0756	0.0398	0.0439	0.0417	0.0534	0.03811
Snímková frekvence výstupu [FPS]	13	25	23	24	19	25

Tabulka 5.1: Porovnání jednotlivých videosouborů použitých pro testování.

5.4 Testovací sestava

Veškeré testování proběhlo na testovací sestavě:

- AMD Athlon II X3 445, 3100Mhz taktovaný na 3797Mhz,
- 12GB RAM,
- operační systém Ubuntu 14.04 64bit.

Kapitola 6

Závěr

V této práci byly popsány různé, v dnešní době používané, metody pro sledování chodců od satelitních systémů, až po ty kamerové. Dále byly vybrány a podrobně popsány metody používané pro zpracování obrazu. Z těchto metod byly vybrány ty nejvhodnější, a na jejich základech byl navržen algoritmus pro detekci a sledování pohybu osob se záznamem trajektorií pohybu.

Pro zjištění osob ve snímku bylo využito detekce popředí pomocí směsice Gaussových křivek, a následně popsání získaných objektů histogramy orientovaných gradientů spolu s algoritmem pro strojové učení SVM. Takto detekované a zpracované osoby pak bylo možné sledovat za použití navrženého algoritmu postaveného na histogramech barevných kanálů.

Výsledný algoritmus byl otestován na množině videí z městského prostředí. Algoritmus je plně schopný detekovat a sledovat jednoho chodce po celou dobu jeho pohybu ve scéně. Navržený algoritmus zvládá detekovat a sledovat i větší množství osob (6 a více chodců) najednou v rámci sledované oblasti. Takové použití má, ale své omezení.

Sledované osoby by neměly mít oděvy stejných barev. Osoby by neměly vstupovat do sledované oblasti ve skupinách. Čím blíže k sobě se budou jednotlivé osoby držet, tím bude sledování horší.

Při porušení těchto omezení, může ve výsledku docházet k tomu, že sledování bude vykazovat chyby a, jednotlivé osoby si mohou mezi sebou přebírat své trajektorie, kvůli chybnému přiřazení do páru.

Navržený algoritmus vykazuje za určitých okolností chybovost, a je tedy v tomto stavu nevhodný pro dlouhodobé a přesné sledování osob. V případě použití pro účel sběru statistických dat, je ale použitelný. Aplikace je schopná detekovat chodce a zaznamenat jejich trajektorii. Ve většině případů, kdy dochází k nesprávnému spárování, jdou tyto osoby v obecně stejném směru. Výsledná trajektorie je tedy použitelná pro analýzu, ve kterých místech se chodci v oblasti pohybují. Pro podobnou statistiku je prezentovaný algoritmus dostačující.

Práce byla také zaměřena na výpočetní rychlost řešení. Za určitých okolností je algoritmus schopen sledovat osoby v reálném čase viz. 5.3, tento bod se tedy podařilo splnit.

Navržený algoritmus stále poskytuje mnoho možností pro rozšíření. Natrénováním modelu pro kombinaci klasifikátoru SVM a HOG z vhodných datových sad, je možné dosáhnout lepších výsledků v podobě více klasifikovaných osob z záznamu. Použitím grafického procesoru pro výpočty je možné podstatně zvýšit rychlost zpracování dat a jednotlivých operací. Za určitých okolností, se některé operace mohou zrychlit až několikanásobně. Pro lepší výsledky sledování osob, by bylo zajímavé použití predikce pohybu. Vhodné rozšíření této práce by mohl být i sběr a uchovávání statistických dat o pohybu osob.

Literatura

- [1] Axis Communications AB: *CCD and CMOS sensor technology [online]*. 2010.
URL http://www.axis.com/files/whitepaper/wp_ccd_cmos_40722_en_1010_lo.pdf
- [2] BAY, H.; ESS, A.; TUYTELAARS, T.; aj.: Speeded-Up Robust Features (SURF). In *Computer Vision and Image Understanding [online]*, 2008 [cit. 2016-05-01], ISSN 10773142, s. 346–359, DOI: 10.1016/j.cviu.2007.09.014.
URL <http://linkinghub.elsevier.com/retrieve/pii/S1077314207001555>
- [3] BAY, H.; FASEL, B.; GOOL, L.: Interactive Museum Guide: Fast and Robust Recognition of Museum Objects. In *Proceedings of the first international workshop on mobile vision [online]*, 2006.
URL http://www.vision.ee.ethz.ch/en/publications/papers/proceedings/eth_biwi_00394.pdf
- [4] Report on the Development of BeiDou Navigation Satellite System. Technická zpráva, China Satellite Navigation Office, 2013.
URL <http://www.beidou.gov.cn/attach/2013/12/26/20131226fed336adf2184d52843d5bf81832e82c.pdf>
- [5] BENEZETH, Y.; JODOIN, P.-M.; EMILE, B.; aj.: Comparative study of background subtraction algorithms. In *Journal of Electronic Imaging, Society of Photo-optical Instrumentation Engineers [online]*, 2010, ISSN 10179909, DOI: 10.1117/1.3456695.
URL <http://electronicimaging.spiedigitallibrary.org/article.aspx?articleid=1100758>
- [6] What is a Sensor? SensorCleaning [online]. 2010, [cit. 2016-05-07].
URL <http://www.sensorcleaning.com/whatisasensor.php>
- [7] CMGLEE: Rolling shutter SMIL. Wikipedia: the free encyclopedia [online]. 2015.
URL http://commons.wikimedia.org/wiki/File:Rolling_shutter_SMIL.svg
- [8] CORTES, C.; VAPNIK, V.: Support-vector networks. *Machine Learning [online]*, ročník 20, č. 3, 1995: s. 273–397, ISSN 1573-0565, DOI: 10.1007/BF00994018.
URL <http://link.springer.com/article/10.1007%2FBF00994018>
- [9] DALAL, N.; TRIGGS, B.: Histograms of oriented gradients for human detection. In *Computer Society Conference on Computer Vision and Pattern Recognition [online]*, IEEE, 2005, ISSN 1063-6919, s. 886–893, DOI: 10.1109/CVPR.2005.177.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1467360

- [10] DEZA, M. M.; DEZA, E.: *Encyclopedia of Distances*. Springer, 2014, ISBN 9783662443415, 94 s.
- [11] DOUGLAS, D. H.; PEUCKER, T. K.: Algorithms for the Reduction of the Number of Points Required to Represent a Digitized Line or its Caricature [online]. In *Classics in Cartography*, John Wiley & Sons, Ltd, 2011, ISBN 9780470669488, s. 177–210, DOI: 10.1002/9780470669488.ch2.
URL <http://doi.wiley.com/10.1002/9780470669488.ch>
- [12] FLIR First Mate MS-224 Ultra-Compact, Handheld, Thermal Imager and Night Vision Camera. [online], OceanMedix. [cit. 2016-05-07].
URL http://www.oceanmedix.com/?_siteid=oceanmedix&_sessid=gxwdonkq&action=sku&sku=d0M-FLIR_MS224
- [13] FREEMAN, W. T.; ADELSON, E.: The design and use of steerable filters. In *IEEE Transactions on Pattern Analysis and Machine Intelligence [online]*, IEEE, 1991, ISSN 0162-8828, s. 891–906, DOI: 10.1109/34.93808.
URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=93808>
- [14] FREUND, Y.; SCHAPIRE, R. E.: A Short Introduction to Boosting. In *Journal of Japanese Society for Artificial Intelligence [online]*, Morgan Kaufmann Publishers Inc., 1999, s. 1401–1406.
URL <http://www.site.uottawa.ca/~stan/csi5387/boost-tut-ppr.pdf>
- [15] FUKUNAGA, K.; HOSTETLER, L.: The Estimation of the Gradient of a Density Function, with Applications in Pattern Recognition. In *IEEE Transactions on Information Theory [online]*, IEEE, 1975, ISSN 0018-9448, s. 32–40, DOI: 10.1109/TIT.1975.1055330.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1055330
- [16] What is Galileo? The European Space Agency [online], 2015 [cit. 2016-05-10].
URL http://www.esa.int/Our_Activities/Navigation/The_future_-_Galileo/What_is_Galileo
- [17] GLONASS finally becomes global. NTV [online]. 2011-10-03, [cit. 2016-05-07].
URL <http://www.ntv.ru/novosti/241055>
- [18] GPS constellation status. U.S. Coast guard Navigation center [online], [cit. 2016-05-07].
URL <http://www.navcen.uscg.gov/?Do=constellationStatus>
- [19] Recommendation ITU-R BT.601-7: Studio encoding parameters of digital television for standard 4:3 and wide-screen 16:9 aspect ratios. [online], 2011, international Telecommunication Union.
URL <http://www.itu.int/rec/R-REC-BT.601/>
- [20] JOHNSON, J.: Analysis of Image Forming Systems. In *Image Intensifier Symposium*, 1958, s. 244–273.
- [21] KALMAN, R. E.: A New Approach to Linear Filtering and Prediction Problems. In *Journal of Basic Engineering [online]*, ASME, 1960, ISSN 00219223, s. 35–45, DOI:

- 10.1115/1.3662552.
 URL <http://fluidsengineering.asmedigitalcollection.asme.org/article.aspx?articleid=1430402>
- [22] LINDEBERG, T.: Image Matching Using Generalized Scale-Space Interest Points. In *Scale Space and Variational Methods in Computer Vision [online]*, Springer Berlin Heidelberg, 2013, ISBN 978-3-642-38267-3, s. 355–367, DOI: 10.1007/978-3-642-38267-3_30.
 URL http://link.springer.com/10.1007/978-3-642-38267-3_30
- [23] LINDEBERG, T.: Scale Selection Properties of Generalized Scale-Space Interest Point Detectors. In *Journal of Mathematical Imaging and Vision [online]*, 2013, ISSN 0924-9907, s. 177–210, DOI: 10.1007/s10851-012-0378-3.
 URL <http://link.springer.com/10.1007/s10851-012-0378-3>
- [24] LOWE, D. G.: Object recognition from local scale-invariant features. In *Proceedings of the Seventh IEEE International Conference on Computer Vision [online]*, IEEE, 1999 [cit. 2016-05-01], ISBN 0769501648, s. 1150–1157, DOI: 10.1109/ICCV.1999.790410.
 URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=790410>
- [25] MORDVINTSEV, A.; ABID, K.: Understanding SVM. [online], 2013.
 URL http://opencv-python-tutroals.readthedocs.org/en/latest/py_tutorials/py_ml/py_svm/py_svm_basics/py_svm_basics.html#svm-understanding
- [26] National Research Council: *The global positioning system: a shared national asset: recommendations for technical improvements and enhancements*. 1995, DOI: 10.17226/4920.
- [27] NovAtel Inc. Canada: *GPS+ Reference Manual [online]*. 2007, OM-20000039, 0I.
 URL <http://www.n4iq.com/lucentgps/mayberelated/GPS-Reference.pdf>
- [28] RAMER, U.: An iterative procedure for the polygonal approximation of plane curves [online]. In *Computer Graphics and Image Processing*, Elsevier Inc., 1972, ISSN 0146664x, s. 177–210, DOI: 10.1016/S0146-664X(72)80017-0.
 URL <http://linkinghub.elsevier.com/retrieve/pii/S0146664X72800170>
- [29] REID, B.: Coarse Gaze Estimation in Visual Surveillance. [online], 2009.
 URL http://www.robots.ox.ac.uk/ActiveVision/Research/Projects/2009bbsenfold_headpose/Datasets/TownCentreXVID.avi
- [30] SAMANA, N.: *Global positioning technologies and performance*. Hoboken, N.J: Wiley-Interscience, 2008, ISBN 978-047-0241-905.
- [31] SENtrace Datasheet. PNI Sensor Corporation [online]. 2016, [cit. 2016-05-07].
 URL http://www.pnicorp.com/wp-content/uploads/SENtraceDatasheet_Path.pdf
- [32] SERRA, J.: *Image analysis and mathematical morphology*. Academic Press, 1982, ISBN 0126372403.

- [33] STAUFFER, C.; GRIMSON, W.: Adaptive background mixture models for real-time tracking. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition [online]*, IEEE Comput. Soc, 1999, ISBN 0769501494, s. 246–252, DOI: 10.1109/CVPR.1999.78463.
URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=784637>
- [34] TAMERSOY, B.: *Background Subtraction: Lecture Notes. [online]*. University of Texas at Austin, 2009.
URL http://www.cs.utexas.edu/~grauman/courses/fall2009/slides/lecture9_background.pdf
- [35] VIOLA, P.; JONES, M.: Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE Computer Society Conference [online]*, IEEE, 2001, ISBN 0-7695-1272-0, ISSN 1063-6919, s. 511–518.
URL http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=990517

Příloha A

Obsah CD

CD obsahuje následnou adresářovou strukturu:

- **Text**
zdrojové texty práce
- **Aplikace**
zdrojové a binární soubory této práce
- **Videa**
Testovací videozáznamy